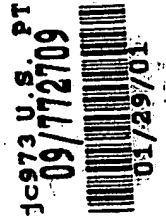


(Translation)

PATENT OFFICE
JAPANESE GOVERNMENT



This is to certify that the annexed is a true copy of
the following application as filed with this Office.

Date of Application: January 31, 2000

Application Number: Japanese Patent Application
No. 22749/2000

Applicant(s): Hitachi Software Engineering Co., Ltd.

January 12, 2001

Commissioner,
Patent Office

Kozo OIKAWA (seal)

Certificate No. 2000-3110497

日 本 国 特 許 庁

PATENT OFFICE
JAPANESE GOVERNMENT

1c973 U.S. PTO
09/772709
01/29/01

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日

Date of Application:

2000年 1月31日

出 願 番 号

Application Number:

特願2000-022749

出 願 人

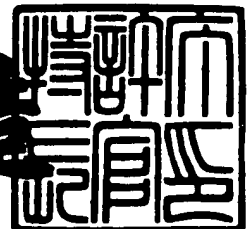
Applicant (s):

日立ソフトウェアエンジニアリング株式会社

2001年 1月12日

特許庁長官
Commissioner,
Patent Office

及 川 耕 造



出証番号 出証特2000-3110497

【書類名】 特許願

【整理番号】 SK11B004

【提出日】 平成12年 1月31日

【あて先】 特許庁長官殿

【国際特許分類】 G06F 17/60

【発明の名称】 インテリジェントなネットワーク中継装置を有するネットワーク構成の自動認識方法及びシステム

【請求項の数】 17

【発明者】

【住所又は居所】 神奈川県横浜市中区尾上町6丁目81番地 日立ソフトウェアエンジニアリング株式会社内

【氏名】 青柳 慶光

【特許出願人】

【識別番号】 000233055

【氏名又は名称】 日立ソフトウェアエンジニアリング株式会社

【代理人】

【識別番号】 100083552

【弁理士】

【氏名又は名称】 秋田 収喜

【電話番号】 03-3893-6221

【手数料の表示】

【予納台帳番号】 014579

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【ブルーフの要否】 要

【書類名】 明細書

【発明の名称】 インテリジェントなネットワーク中継装置を有するネットワーク構成の自動認識方法及びシステム

【特許請求の範囲】

【請求項 1】 SNMP エージェントと管理情報ベースを実装しているインテリジェントなネットワーク機器がネットワークノード内に少なくとも 1 台以上存在するネットワークシステムにおける機器構成を自動認識する方法であって、

SNMP マネージャを実装した管理者端末からネットワークノード内の各ネットワーク機器に対して ICMP エコーリクエストを送信し、その応答によって稼動状態のネットワーク機器を検出する第 1 のステップと、

検出した各ネットワーク機器の SNMP エージェントに対し、当該ネットワーク機器内の管理情報ベースの格納情報の転送要求を送信し、返信された管理情報ベースの格納情報によってネットワークノード内に存在するネットワーク機器の種別を検出する第 2 のステップと

を備えることを特徴とするネットワーク構成の自動認識方法。

【請求項 2】 機器種別がブリッジ機能を有するネットワーク機器の管理情報ベースから当該ネットワーク機器の各ポートに接続されたネットワーク機器の物理アドレスの集合を取得する第 3 のステップと、

ルーティング機能を有するネットワーク機器の管理情報ベースから物理アドレスと IP アドレスの対応情報を取得する第 4 のステップと、

取得した物理アドレスと IP アドレスの対応情報に基づき、ブリッジ機能を有するネットワーク機器の各ポートの接続先のネットワーク機器を IP レベルで認識する第 5 のステップと

をさらに備えることを特徴とする請求項 1 に記載のネットワーク構成の自動認識方法。

【請求項 3】 前記 ICMP エコーリクエストに対して応答が返信されたネットワーク機器は稼動中、応答が返信されないネットワーク機器は存在しないものと認識し、さらに前記第 4 のステップで取得した物理アドレスと IP アドレスの対応情報を参照し、稼動中と認識したネットワーク機器以外の対応情報が存在

する場合には当該ネットワーク機器は非稼動中であるものと認識する第6のステップをさらに備えることを特徴とする請求項2に記載のネットワーク構成の自動認識方法。

【請求項4】 ブリッジ機能またはリピータ機能を有するネットワーク機器の管理情報ベースに当該ネットワーク機器の各ポートの接続先に存在する非稼動中のネットワーク機器の情報が格納されているか否かを調べ、格納されている場合は当該格納情報に基づき非稼動中のネットワーク機器の接続関係を検出するステップをさらに備えることを特徴とする請求項3に記載のネットワーク構成の自動認識方法。

【請求項5】 前記第2のステップで取得したネットワーク機器の管理情報ベースの内容によってブリッジ機能を有するネットワーク機器が複数存在するか否かを検出し、複数の存在を検出した場合には、任意のブリッジ機能を有するネットワーク機器を親機器とし、該親機器の特定ポートの接続先に別のブリッジ機能を有するネットワーク機器が存在するか否かをさらに検出し、存在することを検出した場合には、該ネットワーク機器を子機器とし、その子機器の各ポートの接続先の機器構成を検索し、ブリッジ機能を有するネットワーク機器同士の接続関係をポート単位で認識するステップをさらに備えることを特徴とする請求項2に記載のネットワーク構成の自動認識方法。

【請求項6】 前記子機器と接続している親機器のポートの接続先に存在するネットワーク機器の物理アドレスの集合と、子機器の全ポートから親機器へ接続しているポートを除くポートの接続先に存在するネットワーク機器の物理アドレスの集合の和集合の差分を求め、親機器と子機器の間に存在しているネットワーク機器を認識するステップを有することを特徴とする請求項5に記載のネットワーク構成の自動認識方法。

【請求項7】 前記親機器と子機器の間に複数の機器が存在することを認識した場合に、全機器がルーティング機能、ブリッジ機能、リピータ機能にいずれを保持しているか否かを検出し、いずれも保持していない場合にはノンインテリジェントなネットワーク中継装置が存在するものと予測するステップを備えることを特徴とする請求項6に記載のネットワーク構成の自動認識方法。

【請求項 8】 接続関係を認識した前記親機器と子機器についてそれぞれの管理情報ベース内に保持されている物理アドレスを調べ、親機器の管理情報ベース内に子機器の物理アドレスが保持されていない場合および子機器の管理情報ベース内に親機器の物理アドレスが存在しない場合は、親機器と子機器の特定のポートの接続先に存在する機器の物理アドレスの集合に共通で含まれるような任意の機器を選択し、その選択した機器に対する親機器や子機器の接続ポートを基に親機器と子機器の接続関係を絞り込んで認識するステップを備えることを特徴とする請求項 5 に記載のネットワーク構成の自動認識方法。

【請求項 9】 リピータ機能を有するネットワーク機器の任意のポートにおける最新受信フレームの送信元物理アドレスの更新頻度の値を取得し、この値によって当該任意のポートの接続先に稼動している機器の数を認識するステップと

前記更新頻度の値が「0」または「1」以外の場合には当該任意のポートにおける最新受信フレームの送信元物理アドレスの値を所定時間間隔で取得し、当該任意のポートの接続先に存在する全てのネットワーク機器の物理アドレスを認識することを特徴とする請求項 2 に記載のネットワーク構成の自動認識方法。

【請求項 10】 リピータ機能を有するネットワーク機器の任意のポートにおける最新受信フレームの送信元物理アドレスの更新頻度の値を所定時間間隔で取得し、この値が変化しているか否かによってリピータ機能を有するネットワーク機器が R F C 仕様に準拠しているものか否かを認識するステップをさらに備えることを特徴とする請求項 2 に記載のネットワーク構成の自動認識方法。

【請求項 11】 ブリッジ機能を有するネットワーク機器およびリピータ機能を有するネットワーク機器の管理情報ベースの格納情報によって接続関係を認識できないネットワーク機器について、前記管理者端末によって前記ブリッジ機能を有するネットワーク機器およびリピータ機能を有するネットワーク機器の任意のポートを一時的に閉塞し、閉塞前には I C M P エコーリクエスト packets に対する応答があり、閉塞後には応答がなくなる機器である場合、その機器は当該任意のポートの接続先に存在するものとして認識するステップを備えることを特徴とする請求項 2 に記載のネットワーク構成の自動認識方法。

【請求項 1 2】 ブリッジ機能を有するネットワーク機器およびリピータ機能を有するネットワーク機器の管理情報ベースの格納情報によって接続関係を認識できないネットワーク機器について、前記ブリッジ機能を有するネットワーク機器およびリピータ機能を有するネットワーク機器のポート単位の送受信フレームの統計量を所定時間間隔で収集し、前記ポート単位で任意に設定された統計量の値の範囲内にあるポートの組があれば、このポートの組を接続関係にあるポートとして認識するステップを備えることを特徴とする請求項 2 に記載のネットワーク構成の自動認識方法。

【請求項 1 3】 稼動中のネットワーク機器の管理情報ベースの格納情報を所定時間間隔で収集し、管理者端末の記憶領域に保持し、前回の収集内容と今回の収集内容との相違があるか否かを比較し、稼動中のネットワーク機器の起動、停止、接続先の変更、IP アドレスの変更等を検出するステップを備えることを特徴とする請求項 3 ～ 1 2 のいずれか一項に記載のネットワーク構成の自動認識方法。

【請求項 1 4】 ネットワーク機器同士の接続関係の情報により機器同士の接続関係のモデルテーブルを作成し、前記モデルテーブルを参照して機器同士の接続関係のモデルごとに、または複数の機器同士の接続関係のモデルを組合せることによってネットワーク機器同士の接続関係を検出するステップを備えることを特徴とする請求項 3 ～ 1 3 のいずれか一項に記載のネットワーク構成の自動認識方法。

【請求項 1 5】 認識したネットワーク接続構成を論理的な図面データに展開し、さらに物理的なフロア図面等に物理的な機器構成を配置した図面データを作成し、少なくとも 1 つの図面データを表示装置画面に表示させるステップを備えることを特徴とする請求項 2 ～ 1 3 のいずれか一項に記載のネットワーク構成の自動認識方法。

【請求項 1 6】 SNMP エージェントと管理情報ベースを実装しているインテリジェントなネットワーク機器がネットワークノード内に少なくとも 1 台以上存在するネットワークシステムにおける機器構成を SNMP マネージャを実装した管理者端末から自動認識するシステムであって、

前記 S N M P マネージャを実装した管理者端末が、

ネットワークノード内の各ネットワーク機器に対して I C M P エコーリクエストを送信し、その応答によって稼動状態のネットワーク機器を検出する第 1 の手段と、

検出した各ネットワーク機器の S N M P エージェントに対し、当該ネットワーク機器内の管理情報ベースの格納情報の転送要求を送信し、返信された管理情報ベースの格納情報によってネットワークノード内に存在するネットワーク機器の種別を検出する第 2 の手段と

を備えることを特徴とするネットワーク構成の自動認識システム。

【請求項 1 7】 機器種別がブリッジ機能を有するネットワーク機器の管理情報ベースから当該ネットワーク機器の各ポートに接続されたネットワーク機器の物理アドレスの集合を取得する第 3 の手段と、

ルーティング機能を有するネットワーク機器の管理情報ベースから物理アドレスと I P アドレスの対応情報を取得する第 4 の手段と、

取得した物理アドレスと I P アドレスの対応情報に基づき、ブリッジ機能を有するネットワーク機器の各ポートの接続先の機器を I P レベルで認識する第 5 の手段と

をさらに備えることを特徴とする請求項 1 6 に記載のネットワーク構成の自動認識システム。

【発明の詳細な説明】

【 0 0 0 1 】

【発明の属する技術分野】

本発明は、ルータ、スイッチ、ブリッジ、リピータ、ハブ、端末等が接続されたネットワークに S N M P (Simple Network Management Protocol) を実装しているインテリジェントなネットワーク中継装置を含んでいる場合の各機器の物理的なネットワーク接続構成を自動的に認識する方法およびシステムに関するものである。

【 0 0 0 2 】

【従来の技術】

ルータ、スイッチ、ブリッジ、リピータ、ハブ、端末等が接続されたネットワークにおける各機器の物理的なネットワーク接続構成の認識技術は、ネットワーク監視・管理システムやネットワーク図面作成システム等では必要不可欠な技術である。

従来までのネットワークの接続構成の認識技術では、I P (Intenet Protocol) ネットワークセグメントで分割(ルータで区切られるセグメントを単位とした分割)されたネットワークの認識は可能であった。この技術の範囲では、ルータにより分割されたネットワークセグメントに存在する機器を並列的に羅列するといった表示方法を取ることが多く、実際のネットワークの物理的な構成との対応が取れないという問題があった。

上記の問題を解決するため、「ネットワーク接続装置タイプ検出方法」(特開平 1 1 - 9 6 0 9 4 号)、「ネットワーク監視及びリピータは部の接続端末認識方式」(特開平 1 1 - 1 4 6 0 0 3 号)、「中継装置及びネットワーク管理装置」(特開平 1 0 - 3 3 6 2 2 8 号)、「B G P ルーティング情報を用いたネットワークマップの自動生成方法」(特開平 9 - 1 8 1 7 2 2 号)、「ネットワークトポロジ認識方法およびネットワークトポロジ認識装置」(特開平 9 - 1 8 6 7 1 6 号)、「ネットワーク構成の認識方法」(特開平 8 - 1 9 1 3 2 6 号)の提案がなされている。

【 0 0 0 3 】

【発明が解決しようとする課題】

しかしながら、「ネットワーク接続装置タイプ検出方法」では、機器間のリンクに対して検査パケットを送信することにより、ブリッジとブリッジの接続先の機器との間に存在するループ接続を認識する技術を提供しているが、ループ接続に特化している問題がある。

また、「ネットワーク監視及びリピータハブの接続端末認識方式」では、リピータ M I B (M I B : Management information Base; 管理情報ベース)を利用することでリピータの各ポートの接続先の端末を認識する技術を提供しているが、リピータの各ポートの接続先に複数の端末が接続されている場合には検出できない問題がある。

また、「中継装置及びネットワーク管理装置」では、ネットワーク中継装置の接続関係を検出する技術を提供しているが、特別なハードウェアに依存した実現手段であり、既存のネットワーク構成における実現手段とはなり得ない問題がある。

また、「BGPルーティング情報を用いたネットワークマップの自動生成方法」では、BGP(Border Gateway Protocol;境界ゲートウェイプロトコル)対応ルータに特化したAS(Autonomous System;自立システム)間の接続関係を検出する方法を提供しているが、ネットワークセグメント内の接続関係を識別することはできない問題がある。

また、「ネットワークトポロジ認識方法およびネットワークトポロジ認識装置」では、スパンニングツリープロトコルを用いてブリッジ機器の接続状況を把握する技術を提供しているが、ソースルーティングプロトコルに対応したブリッジ間の接続関係を検出できない問題がある。

また、「ネットワーク構成の認識方法」では、ハブの各ポートの接続先が1台の端末だけであるという条件下で、ハブ(インテリジェントハブ)の各ポートの接続先のMACアドレスの情報をリピータMIBを利用して収集することにより、ポートの接続先にある機器の物理アドレスを把握する技術を提供している。しかし、ハブ同士がカスケード接続している場合にはハブの各ポートの接続先の構成を検出することはできず、各端末から定期的に発信する手段を必要とするため、エージェントソフトウェアを全端末に導入しなくてはならない。また、リピータMIBの実装はベンダによって仕様が異なっており、汎用的なリピータにおける解決方法とは言えない。

【0004】

本発明は、SNMPを実装しているインテリジェントなネットワーク機器が稼動しているネットワーク環境において、SNMP以外の特別なソフトウェアの実装を必要とせず、またSNMPの実装の仕方に依存せずに、少なくとも1台の管理端末からネットワークノード内部の物理的な機器構成を自動的に検出することができるネットワーク構成自動認識方法およびシステムを提供することを目的としている。

【 0 0 0 5 】

【課題を解決するための手段】

本発明は、上記目的を達成するために、基本的には、SNMPエージェントと管理情報ベースを実装しているインテリジェントなネットワーク機器がネットワークノード内に少なくとも1台以上存在するネットワーク環境において、SNMPマネージャを実装した管理者端末からネットワークノード内の各ネットワーク機器に対してICMPエコーリクエストを送信し、その応答によって稼動状態のネットワーク機器を検出する第1のステップと、検出した各ネットワーク機器のSNMPエージェントに対し、当該ネットワーク機器内の管理情報ベースの格納情報の転送要求を送信し、返信された管理情報ベースの格納情報によってネットワークノード内に存在するネットワーク機器の種別を検出する第2のステップとを備えることを特徴とする。

さらに、機器種別がブリッジ機能を有するネットワーク機器の管理情報ベースから当該ネットワーク機器の各ポートに接続されたネットワーク機器の物理アドレスの集合を取得する第3のステップと、ルーティング機能を有するネットワーク機器の管理情報ベースから物理アドレスとIPアドレスの対応情報を取得する第4のステップと、取得した物理アドレスとIPアドレスの対応情報に基づき、ブリッジ機能を有するネットワーク機器の各ポートの接続先の機器をIPレベルで認識する第5のステップとをさらに備えることを特徴とする。

【 0 0 0 6 】

また、前記ICMPエコーリクエストに対して応答が返信されたネットワーク機器は稼動中、応答が返信されないネットワーク機器は存在しないものと認識し、さらに前記第4のステップで取得した物理アドレスとIPアドレスの対応情報を参照し、稼動中と認識したネットワーク機器以外の対応情報が存在する場合には当該ネットワーク機器は非稼動中であるものと認識する第6のステップをさらに備えることを特徴とする。

【 0 0 0 7 】

【発明の実施の形態】

以下、本発明の一実施の形態を図面を用いて詳細に説明する。

図 1 は、本発明を実施するネットワークシステムの一実施形態を示す図である。図示するネットワークは、バックボーンネットワーク 1 を中心に LAN を構築しており、ルータ 2 a、2 b、スイッチングハブ 3、ブリッジ 4、インテリジェントハブ 5、ノンインテリジェントハブ 6 等の中継装置を備えている。これらの中継装置は、“13X.XXX.2.1”のように固有の IP アドレスが割当てられている。

【 0 0 0 8 】

ルータ 2 a (IP アドレス “13X.XXX.2.1”) は、バックボーンネットワーク 1 と内部のセグメントを分割している。すなわち、IP アドレス “13X.XXX.1.*” のネットワークと “13X.XXX.2.*” のネットワークに分割し、“13X.XXX.1.*” 側のネットワークからは IP アドレスが “13X.XXX.1.7” として認識されるが、“13X.XXX.2.*” 側のネットワークからは IP アドレスが “13X.XXX.2.1” として認識されるようになっている。

【 0 0 0 9 】

同様に、ルータ (IP アドレス “13X.XXX.7.1”) 2 b は、バックボーンネットワークと内部のセグメントを分割している。すなわち、IP アドレス “13X.XXX.1.*” のネットワークと “13X.XXX.7.*” のネットワークに分割し、“13X.XXX.1.*” 側のネットワークからは IP アドレスが “13X.XXX.1.9” として認識されるが、“13X.XXX.7.*” 側のネットワークからは IP アドレスが “13X.XXX.7.1” として認識されるようになっている。

【 0 0 1 0 】

内部のセグメントは、スイッチングハブ (IP アドレス “13X.XXX.2.246”) 3 等のスイッチ機器、ブリッジ (IP アドレス “13X.XXX.2.245”) 4、インテリジェントハブ (IP アドレス “13X.XXX.2.243”) 5、ノンインテリジェントハブ (IP アドレスを保持していない) 6 等のネットワーク中継装置を用いてさらに分割されている。

【 0 0 1 1 】

これらのネットワーク中継装置には、別のネットワーク中継装置や端末装置 7 1 ~ 7 8 を接続することで LAN が構築される。

【 0 0 1 2 】

図示するネットワークには、1 台の管理者端末 7 1 が接続されており、この管理者端末 7 1 内でネットワーク構成を自動的に検出するプログラムが稼動している。端末装置 7 2 ～ 7 8 は、稼動中端末装置 7 2 ～ 7 7 と非稼動中端末装置 7 8 に分類可能であり、本実施形態のネットワーク構成自動認識方法では両方を認識対象にしている。

【 0 0 1 3 】

図 1 では、ルータ 2 a とスイッチングハブ 3 を接続し、スイッチングハブ 3 からブリッジ 4、インテリジェントハブ 5、ノンインテリジェントハブ 6 にそれぞれ接続しており、スイッチングハブ 3 に管理者端末 7 1 を接続している例を示している。また、ブリッジ 4 からは 1 台の非稼動中端末装置 7 8 に、インテリジェントハブ 5 やノンインテリジェントハブ 6 からは 3 台の稼動中端末装置 7 2 ～ 7 7 に接続している例を示している。

【 0 0 1 4 】

本実施形態では、1 台の管理者端末 7 1 以外の端末装置 7 2 ～ 7 8 にプログラムを追加することなく、管理者端末 7 1 上にネットワーク構成の自動認識サービスプログラムと図面表示プログラムを追加するだけで機器同士の接続構成を自動的に検出するものである。

なお、前記自動認識サービスプログラムは、SNMP のマネージャとしての機能を備えるものである。また、認識対象のネットワーク機器には SNMP のエージェントを実装したものと実装していないものがある。

【 0 0 1 5 】

初めに、本実施形態のネットワーク構成自動認識方法の概要について説明する。

ネットワーク構成自動認識サービスプログラムは、稼動状況検出モジュール、MIB アクセスモジュール、オートディスカバリモジュールの 3 つのモジュールから構成されている。

稼動状況検出モジュールは、ICMP (Internet Control Message Protocol) エコーリクエストを用いてネットワーク上の各機器の稼動状況を検出するソフトウェアモジュールであり、ICMP エコーリクエストの応答がない IP アドレス

の機器は稼動していないと判断することにより、不要な通信の発生を回避しながらネットワーク上の各機器の稼動状況を検出する機能を備えている。

M I B アクセスモジュールは、S N M P メッセージ (Get-Request PDU、Get-Next PDU、Set-Request PDU) を作成し、S N M P メッセージを送信したり、S N M P メッセージ (Get-Response PDU) を受信して M I B オブジェクトの値を取得する機能を備えたソフトウェアモジュールである。この M I B アクセスモジュールは、各ネットワーク機器が S N M P の M I B オブジェクトを実現していることを前提とするものである。

オートディスカバリモジュールは、ネットワーク構成を検出する機能を備えたソフトウェアモジュールであり、次のようなプロセスによりネットワーク構成を検出する。

- (1) 機器の稼動状況の検出プロセス
- (2) 機器の情報 (I P アドレス、M a c アドレス、ホスト名、サポート M I B、機器種別) の検出プロセス
- (3) M I B オブジェクトの情報の取得プロセス
- (4) ネットワーク中継装置同士の接続関係 (接続ポート) の検出プロセス
- (5) ノンインテリジェントハブの予測プロセス

(1) のプロセスでは、稼動状況検出モジュールを利用して機器の稼動状況を検出する。

(2) のプロセスでは、M I B アクセスモジュールを利用して、実際に M I B にアクセスし、応答が返るかエラーが返るかをチェックすることによって機器がサポートしている M I B を検出する。機器種別は、I P M I B (ipForwarding オブジェクトの値)、ブリッジ M I B サポートの有無、リピータ M I B サポートの有無の情報を組合せることでルータ、ブリッジ、スイッチングハブ、インテリジェントハブ、端末装置、プリンタのいずれに該当するかを分類して検出する (図 1 8 参照)。

(3) のプロセスでは、機器同士の接続関係の検出に利用する M I B オブジェクトの値を取得し、テーブル中に格納する (図 1 4 ～ 図 1 7 参照)。この場合、前記 (1) のプロセスで稼動中でないと判断した機器 (I P アドレス) の情報が M I B オ

プロジェクト中にキャッシュされている場合は、稼動中でない機器の接続情報も取得可能である(図59参照)。

(4)のプロセスでは、上記の機器の中で端末装置を除くネットワーク中継装置同士の接続関係を検出するため、ブリッジMIB、リピータMIB、インタフェースMIBを利用する。

ブリッジMIBは、ネットワーク中継装置の各ポートの接続先の機器のMacアドレスを記憶するオブジェクトを保持しており、各ネットワーク中継装置のポート単位の接続関係を検出可能である。リピータMIBは、ポートの接続先の任意の機器が送信したフレームの中で最後に受信したフレームの送信元のMacアドレスを記憶するオブジェクトを保持しており、所定時間間隔で送信元のMacアドレスを学習することで、各ネットワーク中継装置のポート単位の接続関係を検出可能である。但し、リピータMIBの実装によっては、最後に受信したフレームの送信元のMacアドレスを更新しないネットワーク中継装置が存在しており、上記のリピータMIBを利用してもMacアドレスを学習することができない場合がある。この場合は、インタフェースMIBのポートの状態を変更し、一時的にポートを塞ぎ、ICMPエコーリクエストの応答が返らなくなった機器は塞いだポートの接続先にあると判断する方法や複数のネットワーク中継装置におけるインタフェースMIBのポート単位の送受信フレームの統計量を取得し、統計量に有意な差があるかどうかの検定を行い、有意な差がないポート同士に接続関係があると判断する方法を用いることで、各ネットワーク中継装置のポート単位の接続関係を検出可能である。また、各MIBから取得できるポート単位の接続情報はネットワーク上のすべての機器の接続情報が格納されているとは限らない。ポート単位の接続情報に不備があり、機器同士の接続関係が検出できない場合がある。このような場合は、MIBから取得可能な接続情報によりネットワーク中継装置を複数のネットワーク中継装置モデルに分類し(図21参照)、機器同士の接続関係のモデルを定義し、機器同士の接続関係の検出条件や接続関係の検出可能性を一般化する(図46参照)。この一般化により、ポート単位の接続情報に不備があり、機器同士の接続関係が検出できない場合でも、他の機器との接続関係の情報を組合せ、接続関係の検出条件を満たす場合には、機器同士の接続関

係を検出できるようになる。

また、複数の機器同士の接続関係のモデルを組合せることで、個々の機器同士の接続関係のモデルだけでは検出できない接続関係が検出できる場合がある(図 5 5 参照)。

【 0 0 1 6 】

(5)のプロセスでは、ノンインテリジェントハブの接続を予測するため、ネットワーク中継装置の1つのポートの接続先に複数の機器が接続しているかどうかを検出し、複数の機器が接続されている場合には、ネットワーク中継装置の該当するポートの接続先にはノンインテリジェントハブが少なくとも1台は稼動していると判断する方法により、ノンインテリジェントハブの接続を予測する。

【 0 0 1 7 】

図面表示プログラムは、ネットワーク構成自動認識サービスプログラムで検出したネットワーク構成を画面上に G U I 表示(図 6 2 参照)するプログラムであり、ネットワーク構成を木構造で表示したり、フロア図面上に配置して表示するといった表示形態を採用することができる。

なお、機器の稼動状況の変化やネットワーク構成が変化した場合には、フロア図面もそれに応じて速やかに変更する必要がある。また、機器の稼動状況としては、起動や停止等の変化が挙げられる。また、ネットワーク構成の変化としては、機器の接続先の変更や I P アドレスの変更等が挙げられる。

図面表示プログラムは、オートディスカバリモジュールを用いて M I B オブジェクトの値を定期的にあるいは予め定めたスケジュールに従って不定期に収集し、M I B オブジェクトの値の変化を監視することで機器の稼動状況の変化やネットワーク構成の変化を検出し、自動的にネットワーク構成の変更をネットワーク構成図面に反映し、ユーザに変更を通知する(図 6 0 参照)。

【 0 0 1 8 】

図 2 は、本実施形態において、機器同士の接続関係を検出するために用いる M I B オブジェクトにアクセスするための標準プロトコルである、S N M P のメッセージフォーマットを示す図である。

S N M P メッセージは、S N M P のバージョン番号を格納する Version 2 0 1

、コミュニティ名を格納するCommunity 2 0 2、SNMPのメッセージの本体を格納するPDU (Protocol Data Unit) 2 0 3のフィールドから構成されている。SNMPメッセージは、Get-Request、Get-Next、Get-Response、Set-Request、Trapの5種類のメッセージに分類される。

Get-Request、Get-Nextとは、MIBを有する機器に対してMIBの値を返信するように指示するメッセージであり、これに対してGet-Responseが返信される。

Set-Requestとは、MIBの値を変更するために発行されるメッセージである。また、Trapとは、MIBを有する管理対象の機器で発生した(監視対象とするイベント：重要なイベント)を管理者端末71に自律的に通知するためのメッセージである。

【0019】

本発明のネットワーク構成自動認識方法では、Trap以外のメッセージを利用する。

Get-Request、Get-Next、Get-Response、Set-RequestメッセージのPDUの構成は共通のフォーマットになっており、図2に示すように、メッセージの種類(上記の4種類)を格納するPDU Type 2 0 4、メッセージの一意な識別子を格納するRequestID 2 0 5、エラーメッセージのIDを格納するError Status 2 0 6、メッセージ中のエラー発生箇所を格納するError Index 2 0 7、アクセスするMIBオブジェクトを識別するための情報を格納したリスト2 0 8～2 0 9から構成されている。MIBオブジェクトを識別するための情報を格納したリストの個々の要素は、MIBオブジェクトを一意に識別するための識別子であるOID (Object Identifier)とMIBオブジェクトの値から構成されている。

【0020】

図3は、本実施形態で対象としているインターネットOIDツリーを示す図である。MIBオブジェクト301はネットワーク中継装置の中で木構造として格納されている。この中で、ネットワーク管理で標準となっているMIB2の情報はiso(1)-org(3)-dod(6)-internet(1)-mgmt(2)-MIB-2(2)のノード302に納されており、OIDは"1.3.6.1.2.2"となっている。

本実施形態では、M I B 2 を利用する方法を示している。この他にも各ベンダごとに提供されているベンダ M I B (iso(1)-org(3)-dod(6)-internet(1)-private(4)-enterprise(1)) を利用する方法もあるが、システムの汎用性を高める意味からネットワーク管理の標準プロトコルである M I B 2 を利用するのが望ましい。

【 0 0 2 1 】

図 4 は、本実施形態で対象としている M I B 2 オブジェクトの構成を示す図である。M I B 2 では、現在 5 0 個のオブジェクトが標準化されており、個々のオブジェクトは M I B -2(2) (iso(1)-org(3)-dod(6)-internet(1)-mgmt(2)-MIB-2(2)) の子オブジェクトと 4 0 1 として管理されるようになっている。M I B 2 には system(1)、interfaces(2)、at(3)、ip(4)、ICMP(5)、…、のグループオブジェクトがあるが、本実施形態では太字表示した system(1)、interfaces(2)、ip(4)、dot1dBridge(17)、SNMPDot3RptrMgt(22)、printMIB(43) のグループオブジェクトを利用することで、ネットワーク構成を自動認識する例を示している。

【 0 0 2 2 】

図 5 は、本実施形態で対象としている system グループオブジェクトの構成を示す図である。system グループオブジェクトには、子オブジェクト 5 0 1 として、sysDescr(1)、sysObjecT I D(2)、…、がつながっている。本実施形態では太字表示した sysDescr(1) を利用することで、ネットワーク構成を自動認識する例を示す。sysDescr(1) は、エンティティ(システム)の情報を示すオブジェクトであり、system グループオブジェクトは M I B 2 を実装しているすべての機器で実装することが必須となっているため、M I B 2 のサポート状況を把握するために利用可能である。

【 0 0 2 3 】

図 6 は、本実施形態で対象としている interfaces グループオブジェクトの構成を示す図である。interfaces グループオブジェクトには、子オブジェクト 6 0 1 として、ifNumber(1)、ifTable(2)、…、がつながっている。ifTable(2) はテーブル形式のデータを示しており、ifTable(1) の下に段落分けしてつながっている ifEntry(1) は、ifTable(2) の個々の行を示している。また、ifEntry(1) の下に段

落分けしてつながっているifIndex(1)、ifDescr(2)、…、ifSpecific(22)はifEntry(1)の個々の列を示している。

ネットワーク中継装置では、ifTable(2)の中に、ネットワーク中継装置のインタフェース(ポート)ごとの情報が格納される。今後、すべてのMIBオブジェクトにおけるテーブル形式のデータは上記の規則に従って格納されるものとする。

本実施形態では太字表示したifAdminStatus(7)、ifInOctets(10)、ifInUcastPkts(11)、ifInNUcastPkts(12)、ifInDiscards(13)、ifInErrors(14)、ifOutOctets(16)、ifOutUcastPkts(17)、ifOutNUcastPkts(18)、ifOutDiscards(19)、ifOutErrors(20)を利用することで、ネットワーク構成を自動認識する例を示す。

ifAdminStatus(7)は、インタフェース(ポート)の設定を示すオブジェクトであり、ポートの状態を外部から制御するために利用可能である。

ifInOctets(10)は、インタフェース(ポート)が受信したオクテット数、ifInUcastPkts(11)は上位プロトコルに渡したユニキャストパケット数、ifInNUcastPkts(12)は上位プロトコルに渡した非ユニキャストパケット数、ifInDiscards(13)はエラー以外の理由で廃棄された到着パケット数、ifInErrors(14)はエラーのため上位プロトコルに渡されなかった到着パケット数を示すオブジェクトである。

【 0 0 2 4 】

同様に、ifOutOctets(16)はインタフェース(ポート)が転送したオクテット数、ifOutUcastPkts(17)は上位プロトコルから受信したユニキャストパケット数、ifOutNUcastPkts(18)は上位プロトコルから受信した非ユニキャストパケット数、ifOutDiscards(19)はエラー以外の理由で廃棄された送出パケット数、ifOutErrors(20)はエラーのため転送されなかった送出パケット数を示すオブジェクトである。

ifInOctets(10)～ifOutErrors(20)は、各ポートに対する統計情報を比較して接続関係が存在するポートを検出するために利用可能である。

【 0 0 2 5 】

図 7 は、本実施形態で対象としているipグループオブジェクトの構成を示す図である。ipグループオブジェクトには子オブジェクト701として、ipForwarding(1)、ipDefaultTTL(2)、…、がつながっている。本実施形態では、太字表示し

たipForwarding(1)、ipNetToMediaPhysAddress(2)、ipNetToMediaNetAddress(3)を利用することで、ネットワーク構成を自動認識する例を示す。

ipForwarding(1)は、エンティティ(システム)がIPルーティング機能を保持しているかどうかを示すオブジェクトであり、ネットワーク中継装置がルータであるかどうかを判断するために利用可能である。

ipNetToMediaPhysAddress(2)は、メディア依存の物理アドレスを示すオブジェクトであり、ipNetToMediaNetAddress(3)はメディア依存の物理アドレスに対するIPアドレスを示すオブジェクトである。

ルータ等のネットワーク中継装置では、ipNetToMediaPhysAddress(2)とipNetToMediaNetAddress(3)に接続しているネットワークセグメントのARP(Address Resolution Protocol; IPアドレスからハードウェアアドレスへの変換手順)処理でのキャッシュの情報が格納されるため、セグメントのARPテーブル(MacアドレスとIPアドレスの組合せ)を取得するために利用可能である。

【 0 0 2 6 】

図8は、本実施形態で対象としているdot1dBrdigeグループオブジェクトの構成を示す図である。dot1dBrdigeグループオブジェクトには、子オブジェクト801として、dot1dBase(1)、dot1dStp(2)、…、がつながっている。

本実施形態では太字表示したdot1dTpFdbAddress(1)、dot1dTpFdbPort(2)を利用することで、ネットワーク構成を自動認識する例を示す。

dot1dTpFdbAddress(1)は、ブリッジがforwarding・filtering情報を送信するMACアドレスを示すオブジェクトであり、dot1dTpFdbPort(2)はdot1dTpFdbAddressが送信元アドレスに等しいフレームのポート番号を示すオブジェクトである。ブリッジMIBをサポートしているネットワーク中継装置ではdot1dTpFdbAddress(1)とdot1dTpFdbPort(2)にネットワーク中継装置の各ポートに接続している機器のMacアドレスの集合が格納されるため、ネットワーク中継装置のポート単位の接続機器の情報を取得するために利用可能である。

【 0 0 2 7 】

図9は、本実施形態で対象としているsnmpDot3RptrMgtグループオブジェクトの構成を示す図である。

snmpDot3RpTrMgtグループオブジェクトには子オブジェクト901として、rpTrBasicPackage(1)、rpTrMonitorPackage(2)、…、がつながっている。

本実施形態では、太字表示したrpTrAddrTrackPortIndex(2)、rpTrAddrTrackLastSourceAddress(3)、rpTrAddrTrackSourceAddrChanges(4)を利用することで、ネットワーク構成を自動認識する例を示す。

rpTrAddrTrackPortIndex(2)は、グループに属するポートの識別子を示すオブジェクトであり、rpTrAddrTrackLastSourceAddress(3)は最後に受信したフレームの送信元アドレスを示すオブジェクトであり、rpTrAddrTrackSourceAddrChanges(4)はRpTrAddrTrackLastSourceAddressの変更頻度を示すオブジェクトである。

リピータMIBをサポートしているネットワーク中継装置では、rpTrAddrTrackPortIndex(2)とrpTrAddrTrackLastSourceAddress(3)にネットワーク中継装置の各ポートに接続している機器の任意の1台の機器のMacアドレスが格納される。RFC (Request for Comment) の仕様通りに実装されているネットワーク中継装置では、フレームを受信するたびにrpTrAddrTrackLastSourceAddress(3)の値を更新するため、rpTrAddrTrackLastSourceAddress(3)の情報を学習することでネットワーク中継装置の各ポートに接続している機器のMacアドレスの集合を取得することが可能である。しかし、RFCの仕様通りに実装されていないネットワーク中継装置では、フレームを受信してもrpTrAddrTrackLastSourceAddress(3)の値を更新しない場合がある。ネットワーク中継装置がRFCの仕様通りに実装されているか否かを判断するために、rpTrAddrTrackSourceAddrChanges(4)を利用可能である。

RpTrAddrTrackSourceAddrChanges(4)はrpTrAddrTrackLastSourceAddress(3)の変更頻度を示すため、RFCの仕様通りに実装されているネットワーク中継装置では時間の経過とともに増加していくが、RFCの仕様通りに実装されていないネットワーク中継装置では変動しない。この場合、rpTrAddrTrackSourceAddrChanges(4)には、各ポートの先で検出された機器の数が格納される場合がある。

【0028】

図10は、本実施形態で対象としているprintMIBグループオブジェクトの

構成を示す図である。printMIBグループオブジェクトには、子オブジェクト 1001として、ptrGeneral(5)、ptrGeneralTable(1)、…、がつながっている。本実施形態では、太字表示したptrGeneralConfigChanges(1)を利用することで、ネットワーク構成を自動認識する例を示す。

ptrGeneralConfigChanges(1)は、プリンタの設定変更回数を示すオブジェクトであり、printMIBグループオブジェクトはプリンタで実装されるため、機器がプリンタかどうかを把握するために利用可能である。

【0029】

図11は、管理者端末71に実装されるプログラム構成を示す図である。

図1におけるシステムでは、ネットワーク上の1台の管理者端末71からネットワーク構成を自動認識するため、該管理者端末71には通信ポート1102、ネットワーク構成自動認識サービスプログラム1103、図面表示プログラム1104が実装されている。なお、これらのネットワーク構成自動認識サービスプログラム1103、図面表示プログラム1104は、汎用のコンピュータにインストールして実行可能なようにCD-ROMやDVD-ROM等の記録媒体に記録してユーザに提供することが可能である。また、インターネット等の通信媒体あるいは通信手段を介してユーザに有償で配布することができる。

ネットワーク構成自動認識サービスプログラム1103は、稼動状況検出モジュール1111、MIBアクセスモジュール1112、オートディスカバリモジュール1113から構成されている。

MIBアクセスモジュール1112は、MIB2のOID情報を格納しているOIDテーブル(図12参照)を管理している。

オートディスカバリモジュール1113は、MacアドレスからIPアドレスへのアドレス変換情報を格納しているATテーブル(図13参照)と機器固有の情報を格納しているTIテーブル(図14参照)とネットワーク中継装置のポート単位の接続機器情報を格納しているPFテーブル(図15参照)、ネットワーク構成のTREE構造の接続関係の情報を格納しているTSテーブル(図16参照)を管理している。

【0030】

図 1 2 は、M I B アクセスモジュール 1 1 1 2 が S N M P メッセージ送受信時に利用する O I D (Object Identifier) テーブル 1 1 2 1 の構成を示す図である。

O I D テーブル 1 1 2 1 は、Object Name 1 2 0 1、Object Identifier 1 2 0 2、type 1 2 0 3、Object Path 1 2 0 4 の項目を保持している。

Object Name 1 2 0 1 には M I B アクセスモジュール 1 1 1 2 が O I D テーブル 1 1 2 1 を検索するときのキーとして利用するオブジェクトの一意な名前を格納し、Object Identifier 1 2 0 2 には S N M P メッセージに記述するためのオブジェクトの一意な識別子を格納し、type 1 2 0 3 にはオブジェクトの型を格納し、Object Path 1 2 0 4 にはオブジェクトの完全なパス名を格納する。

M I B アクセスモジュール 1 1 1 2 は、S N M P メッセージ作成時に O I D テーブル 1 1 2 1 にアクセスし、取得する M I B オブジェクトの識別子を検索したり、オブジェクトの型に応じた受信バッファの確保を行う。

【 0 0 3 1 】

図 1 3 は、オートディスカバリモジュール 1 1 1 3 が作成する A T (Address Translation) テーブル 1 1 2 2 の構成を示す図である。

A T テーブル 1 1 2 2 は、IP Address 1 3 0 1、Mac Address 1 3 0 2 の項目を保持している。IP Address 1 3 0 1 には機器の I P アドレス値を格納し、Mac Address 1 3 0 2 には機器の Mac Address 値を格納する。A T テーブル 1 1 2 2 は機器の I P アドレスと M a c アドレスの組の集合を表すため、セグメント全体のアドレス情報をキャッシュしているルータ等の機器から情報を取得して作成する。機器の I P アドレスをキーに M a c アドレスを検索する場合や M a c アドレスから I P アドレスを解決する場合に利用する。

【 0 0 3 2 】

図 1 4 は、オートディスカバリモジュール 1 1 1 3 が作成する T I (Terminal Information) テーブル 1 1 2 3 の構成を示す図である。

T I テーブル 1 1 2 3 は、IP Address 1 4 0 1、Mac Address 1 4 0 2、Host Name 1 4 0 3、type 1 4 0 4、alive 1 4 0 5、mib2 1 4 0 6、forwarding 1 4 0 7、bridge 1 4 0 8、repeater 1 4 0 9、print 1 4 1 0 の項目を保持している。

IP Address 1 4 0 1 には機器の I P アドレス値を格納し、Mac Address 1 4 0 2 には機器の M a c アドレス値を格納し、Host Name 1 4 0 3 には機器のホスト名を格納する。Type 1 4 0 4 には機器の種別を表す識別子を格納する。図 1 4 では、Unknownを表すUに0、Routerを表すRに1、…、Printerを表すPに7が割り当てられるようになっている。

Alive 1 4 0 5 には機器が稼動中か非稼動中かを示すフラグ値を格納する。図 1 4 では、Onに1が、Offに0が割り当てられるようになっている。mib2 1 4 0 6 には機器がmib2をサポートしているか否かを示すフラグ値を格納し、forwarding 1 4 0 7 には機器がIPフォワーディングを行っているか否かを示すフラグ値を格納し、bridge 1 4 0 8 には機器がブリッジ M I B をサポートしているか否かを示すフラグ値を格納し、repeater 1 4 0 9 には機器がリピータ M I B をサポートしているか否かを示すフラグ値を格納する。Print 1 4 1 0 には機器がプリンタ M I B をサポートしているか否かを示すフラグ値を格納する。

オートディスカバリモジュール 1 1 1 3 は T I テーブル 1 1 2 3 を作成することにより、セグメント内で稼動する機器の把握を容易にするとともに、T I テーブル 1 1 2 3 を検索することで機器がサポートする M I B を把握し、非サポート M I B への余計なアクセスを回避することが可能となる。

【 0 0 3 3 】

図 1 5 は、オートディスカバリモジュール 1 1 1 3 が作成する P F (Port Forwarding) テーブル 1 1 2 4 の構成を示す図である。

P F テーブル 1 1 2 4 は、Source IP Address 1 5 0 1、Source Mac Address 1 5 0 2、Source Port 1 5 0 3、Destination IP Address 1 5 0 4、Destination Mac Address 1 5 0 5 の項目を保持している。

Source IP Address 1 5 0 1 にはネットワーク中継装置の I P アドレス値を格納し、Source Mac Address 1 5 0 2 にはネットワーク中継装置の M a c アドレス値を格納し、Source Port 1 5 0 3 にはネットワーク中継装置のポート番号を格納する。

また、Destination IP Address 1 5 0 4 には Source Port 1 5 0 3 のポートの接続先で稼動する機器の I P アドレス値、Destination Mac Address 1 5 0 5 に

はDestination IP Address 1 5 0 4 の機器のMacアドレス値を格納する。PFテーブル 1 1 2 4 は、セグメント内で稼動するネットワーク中継装置と別のネットワーク中継装置または端末装置との接続情報を表す。

【 0 0 3 4 】

図 1 6 は、オートディスカバリモジュール 1 1 1 3 が作成するTS (Tree Structure) テーブル 1 1 2 5 の構成を示す図である。

TS テーブル 1 1 2 5 は、Terminal IP Address 1 6 0 1、Terminal Mac Address 1 6 0 2、Terminal Port 1 6 0 3、Parent IP Address 1 6 0 4、Parent Mac Address 1 6 0 5、Parent Port 1 6 0 6 の項目を保持している。

Terminal IP Address 1 6 0 1 には稼動中の機器のIPアドレス値を格納し、Terminal Mac Address 1 6 0 2 にはIPアドレスがTerminal IP Address 1 6 0 1 の機器のMacアドレス値を格納し、Terminal Port 1 6 0 3 には機器の接続ポート番号を格納する。機器が端末装置の場合やネットワーク中継装置であってポート番号が未知である場合はTerminal Port 1 6 0 3 にNULL値を格納する。また、Parent IP Address 1 6 0 4 にはポート番号がTerminal Port 1 6 0 3 のポートに直接接続しているネットワーク中継装置のIPアドレス値を格納し、Parent Mac Address 1 6 0 5 にはParent IP Address 1 6 0 4 のネットワーク中継装置のMacアドレス値を格納し、Parent Port 1 6 0 6 には接続ポート番号を格納する。

TS テーブル 1 1 2 5 とPF テーブル 1 1 2 4 の違いは、PF テーブル 1 1 2 4 ではネットワーク中継装置の任意のポートの接続先で稼動するすべての機器の情報を格納しているため、1台の機器が複数のネットワーク中継装置のエントリに追加されることがあるが、TS テーブル 1 1 2 5 では1台の機器に対して直接接続しているネットワーク中継装置の情報だけが追加されるということである。

【 0 0 3 5 】

図 1 7 は、MIB アクセスモジュール 1 1 1 2 がSNMPメッセージを送受信する仕組みを示す図である。

管理者端末 7 1 上で稼動するMIB アクセスモジュール 1 1 1 2 は、SNMP メッセージ (Get-Request メッセージまたはGet-Next メッセージ) を作成し、情報

を取得したいネットワーク中継装置（または端末、プリンタ等の機器）1703
上で稼動しているSNMPエージェント1704に対してSNMPメッセージを
送信する。SNMPエージェント1704は、SNMPメッセージを受信すると
該SNMPメッセージを解釈し、要求されているMIBオブジェクトの値を格納
したSNMPメッセージ(Get-Response)を作成し、MIBアクセスモジュール1
112にSNMPメッセージを返送する。これにより、MIBアクセスモジュール1112はネットワーク中継装置1703の任意のMIBオブジェクトの値を
取得することができる。

【0036】

図18は、機器種別の検出方法を説明する図である。

本実施形態でネットワーク構成の認識対象としている機器は、ルータ1801
、ブリッジ1802、スイッチングハブ1803、インテリジェントハブ1804
、ノンインテリジェントハブ1805、プリンタ1806、端末装置1807
である。ルータ1801はipグループのipForwardingオブジェクトの値が“1”
であり、ブリッジMIBを実装しているが、リピータMIBやプリンタMIBを
実装していない機器を示す。

ブリッジ1802はipグループのipForwardingオブジェクトの値が“0”であ
り、ブリッジMIBを実装しているが、リピータMIBやプリンタMIBを実装
していない機器を示す。

スイッチングハブ1803はipグループのipForwardingオブジェクトの値が“
1”または“0”であり、ブリッジMIBとリピータMIBは実装しているが、
プリンタMIBを実装していない機器を示す。

インテリジェントハブ1804はipグループのipForwardingオブジェクトの値
が“0”であり、リピータMIBは実装しているが、ブリッジMIBとプリンタ
MIBを実装していない機器を示す。

ノンインテリジェントハブ1805はMIBを実装していない機器を示す。プ
リンタ1806はipグループのipForwardingオブジェクトの値が“0”であり、
プリンタMIBは実装しているがブリッジMIBとリピータMIBを実装してい
ない機器を示す。

端末装置 1 8 0 7 は ip グループの ipForwarding オブジェクトの値が “0” であり、ブリッジ M I B とリピータ M I B とプリンタ M I B を実装していない機器を示す。

ip グループの ipForwarding オブジェクトの値とブリッジ M I B の実装状況とリピータ M I B の実装状況とプリンタ M I B の実装状況の組合せはどの機器種別においても異なっているため、この組み合わせを調べることによって機器種別の検出が可能となる。

【 0 0 3 7 】

図 1 9 は、ネットワーク中継装置間の Relation 定義を説明する図である。

図 1 9 では異なる 4 台のネットワーク中継装置の親子関係を表しており、バックボーンネットワークと接続し、セグメントの末端のネットワーク中継装置を Root 装置 (I P アドレスは “13X.XXX.2.1”) 1 9 0 1 と定義する。この Root 装置 1 9 0 1 の Port1 の接続先には 3 台のネットワーク中継装置が稼動しており、Root 装置 1 9 0 1 の Port1 と直接接続しているネットワーク中継装置を Parent 装置 (I P アドレスは “13X.XXX.2.246”) 1 9 0 2、この Parent 装置 1 9 0 2 の Port2 の接続先のネットワーク中継装置を Child1 装置 (I P アドレスは “13X.XXX.2.243”) 1 9 0 3、Parent 装置 1 9 0 2 の Port3 の接続先のネットワーク中継装置を Child2 装置 (I P アドレスは “13X.XXX.2.245”) 1 9 0 4 とする。そして、任意のネットワーク中継装置と、接続先に Root 装置が存在するポートとは別のポートの接続先で稼動する任意のネットワーク中継装置は親子であると定義する。

【 0 0 3 8 】

図 1 9 の例では Root 装置 1 9 0 1 と Parent 装置 1 9 0 2、Child1 装置 1 9 0 3、Child2 装置 1 9 0 4 は親子である。また、Parent 装置 1 9 0 2 と Child1 装置 1 9 0 3、Child2 装置 1 9 0 4 は親子である。

また、任意のネットワーク中継装置と、接続先に Root 装置が存在するポートの接続先で稼動する任意のネットワーク中継装置の中で、Root 装置へのホップ数が同一であるネットワーク中継装置の集合は兄弟であると定義する。

図 1 9 の例では Child1 装置 1 9 0 3 の Port1 の接続先には Root 装置 1 9 0 1、Parent 装置 1 9 0 2、Child2 装置 1 9 0 4 が稼動中であり、Child1 装置 1 9 0 3

からRoot装置1901へのホップ数は“1”である。また、Child2装置1904からRoot装置1901へのホップ数は“1”であることから、Child1装置1903とChild2装置1904は兄弟である。

【0039】

図20は本実施形態のinterfaces MIBを利用したネットワーク中継装置間の接続検出方法を説明する図である。図の例のように、異なる2台のネットワーク中継装置Unit1装置(IPアドレスは“13X.XXX.2.246”)2001とUnit2装置(IPアドレスは“13X.XXX.2.243”)2002が稼動している場合、ネットワーク中継装置の各ポートにおけるinterfaces MIBのifInOctetsオブジェクトの値とifOutOctetsオブジェクトの値を同時に取得する。

図20の例ではUnit1装置2001のPort1のifInOctetsオブジェクトの値2003、ifOutOctetsオブジェクトの値2004、Unit2装置2002のifInOctetsオブジェクトの値2005、ifOutOctetsオブジェクトの値2006をそれぞれ取得したことを示している。

Unit1装置2001のPort1のifInOctetsオブジェクトの値2003とUnit2装置2002のifOutOctetsオブジェクトの値2006またはUnit1装置2001のPort1のifOutOctetsオブジェクトの値2004とUnit2装置2002のifInOctetsオブジェクトの値2005)の差の検定を行い、有意な差がないことを算出した場合、Unit1装置2001のPort1とUnit2装置2002のPort1の間に接続関係があることを検出する。ここで、有意の差とは、1例として、2つの値の差がある閾値を超えるような場合には2つの値は異なるといった、2つの値が統計的に異なることを示すものとする。

【0040】

図21は本実施形態におけるネットワーク機器の分類方法を示す図である。

本実施形態でのネットワーク機器モデルは、R2101、CF2102、IF2103、SF2104、NF2105、Term2106である。

Rはセグメントに分割するネットワーク中継装置(Router)を示し、セグメントのすべての機器に対する親となる。また、ネットワーク中継装置はMIBから取得できる機器の接続情報によってCFとIFとSFに分類する。CFはMIBのオブジェ

クトの格納情報に不備がなく、すべてのネットワーク中継装置と端末装置の接続ポートを格納したPFテーブル（図15）を作成可能なネットワーク中継装置を示す。

IFはMIBのオブジェクトの格納情報に不備があり、Rを除く他のネットワーク中継装置への接続ポート番号を検出できない場合が存在するネットワーク中継装置を示す。

SFはMIBのオブジェクトの格納情報に不備があり、Rを含む他のすべてのネットワーク中継装置への接続ポートが検出できず、1台以上の端末装置への接続ポートが検出可能なネットワーク中継装置を示す。また、MIBを実装していないノンインテリジェントハブやリピータをNFとする。プリンタや端末装置等のネットワーク中継装置以外の機器はTermとする。

【0041】

図22は本実施形態のR-CF-CFモデルの接続検出の仕組みを示す。

図22はR-CF-CFモデルの一例として、R(IPアドレス "13X.XXX.2.1") 2201のポート2とCF1(IPアドレス "13X.XXX.2.246") 2202のポート2に接続関係があり、CF1(IPアドレス "13X.XXX.2.246") 2202のポート1とCF2(IPアドレス "13X.XXX.2.243") 2203のポート1に接続関係がある場合を示す。

【0042】

図23は本実施形態のR-CF-CFモデルの接続検出に利用するPFテーブル1124のエントリ例を示すものであり、ここでは図22に示したR-CF-CFモデルに対するPFテーブル1124のエントリを例示している。

ここでは、エントリ2301にCF1(IPアドレス "13X.XXX.2.246") 2202からCF2(IPアドレス "13X.XXX.2.243") 2203への接続情報が格納されている。この接続情報によって、CF12202からCF22203への接続ポートがポート1であることを検出可能である。

同様に、エントリ2302にCF1(IPアドレス "13X.XXX.2.246") 2202からR(IPアドレス "13X.XXX.2.1") 2201への接続情報が格納されている。この接続情報によって、CF12202からR2201への接続ポートがポート2であることを検出可能である。また、CF12202からR2201への接続ポートとCF

12202からCF22203への接続ポートが異なることから、CF12202はCF22203の親であることが検出可能である。

さらに、エントリ2303にCF2(IPアドレス "13X.XXX.2.243")2203からR(IPアドレス "13X.XXX.2.1")2201への接続情報が格納されている。この接続情報によって、CF22203からR2201への接続ポートがポート1であることを検出可能である。

また、エントリ2304にCF2(IPアドレス "13X.XXX.2.243")2203からCF1(IPアドレス "13X.XXX.2.246")2202への接続情報が格納されている。この接続情報によって、CF22203からCF12202への接続ポートがポート1であることを検出可能である。

このようにして、R-CF-CFモデルでは、任意の条件下で機器の接続ポートと親子関係の検出が可能である。

【0043】

図24は本実施形態のR-CF-IFモデルの接続検出の仕組みを示す。

図24はR-CF-IFモデルの一例として、R(IPアドレス "13X.XXX.2.1")2401のポート2とCF(IPアドレス "13X.XXX.2.246")2402のポート2に接続関係があり、CF(IPアドレス "13X.XXX.2.246")2402のポート1とIF(IPアドレス "13X.XXX.2.243")2403のポート1に接続関係がある場合を示している。

【0044】

図25は本実施形態のR-CF-IFモデルの接続検出に利用するPFテーブル1124のエントリ例を示すものであり、ここでは図24に示したR-CF-IFモデルに対するPFテーブル1124のエントリを例示している。

ここでは、エントリ2501にCF(IPアドレス "13X.XXX.2.246")2402からIF(IPアドレス "13X.XXX.2.243")2403への接続情報が格納されている。この接続情報によって、CF2402からIF2403への接続ポートがポート1であることを検出可能である。

また、エントリ2502にCF(IPアドレス "13X.XXX.2.246")2402からR(IPアドレス "13X.XXX.2.1")2401への接続情報が格納されている。この接続情報によって、CF2402からR2401への接続ポートがポート2であること

を検出可能である。また、CF 2 4 0 2 から R 2 4 0 1 への接続ポートと CF 2 4 0 2 から IF 2 4 0 3 への接続ポートが異なることから、CF 2 4 0 2 は IF 2 4 0 3 の親であることが検出可能である。

また、エントリ 2 5 0 3 に IF (IP アドレス "13X.XXX.2.243") 2 4 0 3 から R (IP アドレス "13X.XXX.2.1") 2 4 0 1 への接続情報が格納されている。この接続情報によって、IF 2 4 0 3 から R 2 4 0 1 への接続ポートがポート 1 であることを検出可能である。また、IF 2 4 0 3 から R 2 4 0 1 への接続ポートはポート 1 であり、CF 2 4 0 2 は IF 2 4 0 3 の親であるため、IF 2 4 0 3 から CF 2 4 0 2 への接続ポートは IF 2 4 0 3 から R 2 4 0 1 への接続ポートと等しい。従って、IF 2 4 0 3 から CF 2 4 0 2 への接続ポートはポート 1 であることを検出可能である。

このようにして R-CF-IF モデルでは、任意の条件下で機器の接続ポートと親子関係の検出が可能である。

【 0 0 4 5 】

図 2 6 は、本実施形態の R-CF-SF モデルの接続検出の仕組みを示す図であり、ここでは、R-CF-SF モデルの一例として、R (IP アドレス "13X.XXX.2.1") 2 6 0 1 のポート 2 と CF (IP アドレス "13X.XXX.2.246") 2 6 0 2 のポート 2 に接続関係があり、CF (IP アドレス "13X.XXX.2.246") 2 6 0 2 のポート 1 と SF (IP アドレス "13X.XXX.2.243") 2 6 0 3 のポート 1 に接続関係があり、CF (IP アドレス "13X.XXX.2.246") 2 6 0 2 のポート 3 の先に任意の Term1 (IP アドレス "13X.XXX.2.102") 2 6 0 4 が接続されている場合を示している。

【 0 0 4 6 】

図 2 7 は本実施形態の R-CF-SF モデルの接続検出に利用する P F テーブル 1 1 2 4 のエントリ例を示すものであり、図 2 6 に示した R-CF-SF モデルに対する P F テーブル 1 1 2 4 のエントリを例示している。

エントリ 2 7 0 1 に CF (IP アドレス "13X.XXX.2.246") 2 6 0 2 から SF (IP アドレス "13X.XXX.2.243") 2 6 0 3 への接続情報が格納されている。この接続情報によって、CF 2 6 0 2 から SF 2 6 0 3 への接続ポートがポート 1 であることを検出可能である。

また、エントリ 2 7 0 2 に CF (IP アドレス "13X.XXX.2.246") 2 6 0 2 から R (

I P アドレス "13X.XXX.2.1") 2 6 0 1 への接続情報が格納されている。この接続情報によって、CF 2 6 0 2 から R 2 6 0 1 への接続ポートがポート 2 であることを検出可能である。また、CF 2 6 0 2 から R 2 6 0 1 への接続ポートと CF 2 6 0 2 から SF 2 6 0 3 への接続ポートが異なることから、CF 2 6 0 2 は SF 2 6 0 3 の親であることが検出可能である。

また、エントリ 2 7 0 3 に CF (I P アドレス "13X.XXX.2.246") 2 6 0 2 から Term1 (I P アドレス "13X.XXX.2.102") 2 6 0 4 への接続情報が格納されている。この接続情報によって、CF 2 6 0 2 から Term1 2 6 0 4 への接続ポートがポート 3 であることを検出可能である。また、CF 2 6 0 2 から SF 2 6 0 3 への接続ポートと CF 2 6 0 2 から Term1 2 6 0 4 への接続ポートが異なることから、Term1 2 6 0 4 は SF 2 6 0 3 に接続している機器ではないことが検出可能である。

また、エントリ 2 7 0 4 に SF (I P アドレス "13X.XXX.2.243") 2 6 0 3 から Term1 (I P アドレス "13X.XXX.2.102") 2 6 0 4 への接続情報が格納されている。この接続情報によって、SF 2 6 0 3 から Term1 2 6 0 4 への接続ポートがポート 1 であることを検出可能である。また、SF 2 6 0 3 から Term1 2 6 0 4 への接続ポートはポート 1 であり、Term1 2 6 0 4 は SF 2 6 0 3 に接続している機器ではないことから、SF 2 6 0 3 から CF 2 6 0 2 への接続ポートは SF 2 6 0 3 から Term1 2 6 0 4 への接続ポートと等しい。

従って、SF 2 6 0 3 から CF 2 6 0 2 への接続ポートはポート 1 であることを検出可能である。

このようにして、R-CF-SF モデルでは、CF 2 6 0 2 と Term1 2 6 0 4 の接続情報と SF 2 6 0 3 と Term1 2 6 0 4 の接続情報が P F テーブル 1 1 2 4 に格納されているという条件下で機器の接続ポートと親子関係の検出が可能である。

【 0 0 4 7 】

図 2 8 は本実施形態の R-IF-CF モデルの接続検出の仕組みを示す図であり、R-IF-CF モデルの一例として、R (I P アドレス "13X.XXX.2.1") 2 8 0 1 のポート 2 と IF (I P アドレス "13X.XXX.2.246") 2 8 0 2 のポート 2 に接続関係があり、IF (I P アドレス "13X.XXX.2.246") 2 8 0 2 のポート 1 と CF (I P アドレス "13X.XXX.2.243") 2 8 0 3 のポート 1 に接続関係があり、CF (I P アドレス "13X.XXX.2.243")

2803のポート2の先に任意のTerm1(IPアドレス "13X.XXX.2.2") 2804が接続されている場合を例示している。

【0048】

図29は本実施形態のR-IF-CFモデルの接続検出に利用するPFテーブル1124のエントリ例を示すものであり、図28に示したR-IF-CFモデルに対するPFテーブル1124のエントリを例示している。

ここでは、エントリ2901にIF(IPアドレス "13X.XXX.2.246") 2802からTerm1(IPアドレス "13X.XXX.2.2") 2804への接続情報が格納されている。この接続情報によって、IF2802からTerm12804への接続ポートがポート1であることを検出可能である。

また、エントリ2902にIF(IPアドレス "13X.XXX.2.246") 2802からR(IPアドレス "13X.XXX.2.1") 2801への接続情報が格納されている。この接続情報によって、IF2802からR2801への接続ポートがポート2であることを検出可能である。

また、エントリ2903にCF(IPアドレス "13X.XXX.2.243") 2803からR(IPアドレス "13X.XXX.2.1") 2801への接続情報が格納されている。この接続情報によって、CF2803からR2801への接続ポートがポート1であることを検出可能である。

また、エントリ2904にCF(IPアドレス "13X.XXX.2.243") 2803からIF(IPアドレス "13X.XXX.2.246") 2802への接続情報が格納されている。この接続情報によって、CF2803からIF2802への接続ポートがポート1であることを検出可能である。また、CF2803からR2801への接続ポートとCF2803からIF2802への接続ポートが等しいことから、IF2802はCF2803の親である、またはIF2802とCF2803は兄弟であることを検出可能である。

また、エントリ2905にCF(IPアドレス "13X.XXX.2.243") 2803からTerm1(IPアドレス "13X.XXX.2.2") 2804への接続情報が格納されている。この接続情報によって、CF2803からTerm12804への接続ポートがポート1であることを検出可能である。また、CF2803からR2801への接続ポートとC

F2 8 0 3 から Term1 2 8 0 4 への接続ポートが異なることから、Term1 2 8 0 4 は CF2 8 0 3 に接続している機器であり、IF2 8 0 2 から R2 8 0 1 への接続ポートと IF2 8 0 2 から Term1 2 8 0 4 への接続ポートが異なる。従って、IF2 8 0 2 は CF2 8 0 3 の親であり、IF2 8 0 2 から CF2 8 0 3 への接続ポートはポート1であることが検出可能である。

このようにして、R-IF-CFモデルでは、IF2 8 0 2 と Term1 2 8 0 4 の接続情報と CF2 8 0 3 と Term1 2 8 0 4 の接続情報が PF テーブル 1 1 2 4 に格納されているという条件下で機器の接続ポートと親子関係の検出が可能である。

【 0 0 4 9 】

図 3 0 は本実施形態の R-IF-IF モデルの接続検出の仕組みを示す図であり、R-IF-IF モデルの一例として、R (IP アドレス "13X.XXX.2.1") 3 0 0 1 のポート 2 と IF1 (IP アドレス "13X.XXX.2.246") 3 0 0 2 のポート 2 に接続関係があり、IF1 (IP アドレス "13X.XXX.2.246") 3 0 0 2 のポート 1 と IF2 (IP アドレス "13X.XXX.2.243") 3 0 0 3 のポート 1 に接続関係があり、さらに IF1 (IP アドレス "13X.XXX.2.246") 3 0 0 2 のポート 3 の先に任意の Term1 (IP アドレス "13X.XXX.2.102") 3 0 0 4 が接続されており、IF2 (IP アドレス "13X.XXX.2.243") 3 0 0 3 のポート 2 の先に任意の Term2 (IP アドレス "13X.XXX.2.2") 3 0 0 5 が接続されている場合を示している。

【 0 0 5 0 】

図 3 1 は本実施形態の R-IF-IF モデルの接続検出に利用する PF テーブル 1 1 2 4 のエントリ例を示すものであり、図 3 0 に示した R-IF-IF モデルに対する PF テーブル 1 1 2 4 のエントリを例示している。

ここでは、エントリ 3 1 0 1 に IF1 (IP アドレス "13X.XXX.2.246") 3 0 0 2 から Term2 (IP アドレス "13X.XXX.2.2") 3 0 0 5 への接続情報が格納されている。この接続情報によって、IF1 3 0 0 2 から Term2 3 0 0 5 への接続ポートがポート 1 であることを検出可能である。

また、エントリ 3 1 0 2 に IF1 (IP アドレス "13X.XXX.2.246") 3 0 0 2 から R (IP アドレス "13X.XXX.2.1") 3 0 0 1 への接続情報が格納されている。この接続情報によって、IF1 3 0 0 2 から R 3 0 0 1 への接続ポートがポート 2 であるこ

とを検出可能である。

また、エントリ 3 1 0 3 に IF1 (IP アドレス "13X.XXX.2.246") 3 0 0 2 から Term1 (IP アドレス "13X.XXX.2.102") 3 0 0 4 への接続情報が格納されている。この接続情報によって、IF1 3 0 0 2 から Term1 3 0 0 4 への接続ポートがポート 3 であることを検出可能である。

また、エントリ 3 1 0 4 に IF2 (IP アドレス "13X.XXX.2.243") 3 0 0 3 から R (IP アドレス "13X.XXX.2.1") 3 0 0 1 への接続情報が格納されている。この接続情報によって、IF2 3 0 0 3 から R 3 0 0 1 への接続ポートがポート 1 であることを検出可能である。

【 0 0 5 1 】

また、エントリ 3 1 0 5 に IF2 (IP アドレス "13X.XXX.2.243") 3 0 0 3 から Term1 (IP アドレス "13X.XXX.2.102") 3 0 0 4 への接続情報が格納されている。この接続情報によって、IF2 3 0 0 3 から Term1 3 0 0 4 への接続ポートがポート 1 であることを検出可能である。

また、エントリ 3 1 0 6 に IF2 (IP アドレス "13X.XXX.2.243") 3 0 0 3 から Term2 (IP アドレス "13X.XXX.2.2") 3 0 0 5 への接続情報が格納されている。この接続情報によって、IF2 3 0 0 3 から Term2 3 0 0 5 への接続ポートがポート 2 であることを検出可能である。また、IF1 3 0 0 2 から R 3 0 0 1 への接続ポートと IF1 3 0 0 2 から Term1 3 0 0 4 への接続ポートが異なることから、IF1 3 0 0 2 は R 3 0 0 1 と Term1 3 0 0 4 の中間に接続している機器であることが検出可能である。

【 0 0 5 2 】

同様に IF2 3 0 0 3 から R 3 0 0 1 への接続ポートと IF2 3 0 0 3 から Term1 3 0 0 4 への接続ポートが等しいことから、Term1 3 0 0 4 は R 3 0 0 1 と IF2 3 0 0 3 の中間に接続している機器であることが検出可能である。従って、IF1 3 0 0 2 は R 3 0 0 1 と IF2 3 0 0 3 の中間に接続している機器であることが検出可能であり、IF1 3 0 0 2 は IF2 3 0 0 3 の親であることが検出可能である。また、IF1 3 0 0 2 は IF2 3 0 0 3 の親であることから、IF2 3 0 0 2 から R 3 0 0 1 への接続ポートと IF2 3 0 0 3 から IF1 3 0 0 2 への接続ポートは等しいため、IF2 3 0

03からIF13002への接続ポートはポート1であることが検出可能である。また、IF23003からR3001への接続ポートとIF23003からTerm23005への接続ポートが異なることから、IF23003はR3001とTerm23005の中間に接続している機器であることが検出可能である。また、IF13002はIF23003の親であることから、IF23003はIF13002とTerm23005の中間に接続しているため、IF13002からTerm23005への接続ポートとIF13002からIF23003への接続ポートは等しい。

従って、IF13002からIF23003への接続ポートはポート1であることが検出可能である。

このようにして、R-IF-IFモデルでは、IF13002とTerm13004、Term23005の接続情報とIF23003とTerm13004、Term23005の接続情報がPFテーブル1124に格納されているという条件下で機器の接続ポートと親子関係の検出が可能である。

【0053】

図32は、本実施形態のR-IF-SFモデルの接続検出の仕組みを示す図であり、R-IF-SFモデルの一例として、R(IPアドレス "13X.XXX.2.1") 3201のポート2とIF(IPアドレス "13X.XXX.2.246") 3202のポート2に接続関係があり、IF(IPアドレス "13X.XXX.2.246") 3202のポート1とSF(IPアドレス "13X.XXX.2.243") 3203のポート1に接続関係があり、IF(IPアドレス "13X.XXX.2.246") 3202のポート3の先に任意のTerm1(IPアドレス "13X.XXX.2.102") 3204が接続されており、SF(IPアドレス "13X.XXX.2.243") 3203のポート2の先に任意のTerm2(IPアドレス "13X.XXX.2.2") 3205が接続されており、さらにSF(IPアドレス "13X.XXX.2.243") 3203のポート3の先に任意のTerm3(IPアドレス "13X.XXX.2.110") 3206が接続されている場合を示している。

【0054】

図33は、本実施形態のR-IF-SFモデルの接続検出に利用するPFテーブル1124のエントリ例を示すものであり、図32に示したR-IF-SFモデルに対するPFテーブル1124のエントリを例示している。

エントリ3301にIF(IPアドレス "13X.XXX.2.246") 3202からTerm2(I

P アドレス "13X.XXX.2.2") 3 2 0 5 への接続情報が格納されている。この接続情報によって、IF 3 2 0 2 からTerm2 3 2 0 5 への接続ポートがポート1であることを検出可能である。

また、エントリ 3 3 0 2 にIF(I P アドレス "13X.XXX.2.246") 3 2 0 2 からTerm3(I P アドレス "13X.XXX.2.110") 3 2 0 6 への接続情報が格納されている。この接続情報によって、IF 3 2 0 2 からTerm3 3 2 0 6 への接続ポートがポート1であることを検出可能である。

また、エントリ 3 3 0 3 にIF(I P アドレス "13X.XXX.2.246") 3 2 0 2 からR(I P アドレス "13X.XXX.2.1") 3 2 0 1 への接続情報が格納されている。この接続情報によって、IF 3 2 0 2 からR 3 2 0 1 への接続ポートがポート2であることを検出可能である。

【 0 0 5 5 】

また、エントリ 3 3 0 4 にIF(I P アドレス "13X.XXX.2.246") 3 2 0 2 からTerm1(I P アドレス "13X.XXX.2.102") 3 2 0 4 への接続情報が格納されている。この接続情報によって、IF 3 2 0 2 からTerm1 3 2 0 4 への接続ポートがポート3であることを検出可能である。

また、エントリ 3 3 0 5 にSF(I P アドレス "13X.XXX.2.246") 3 2 0 3 からTerm1(I P アドレス "13X.XXX.2.102") 3 2 0 4 への接続情報が格納されている。この接続情報によって、SF 3 2 0 3 からTerm1 3 2 0 4 への接続ポートがポート1であることを検出可能である。

また、エントリ 3 3 0 6 にSF(I P アドレス "13X.XXX.2.243") 3 2 0 3 からTerm2(I P アドレス "13X.XXX.2.2") 3 2 0 5 への接続情報が格納されている。この接続情報によって、SF 3 2 0 3 からTerm2 3 2 0 5 への接続ポートがポート2であることを検出可能である。

また、エントリ 3 3 0 7 にSF(I P アドレス "13X.XXX.2.243") 3 2 0 3 からTerm3(I P アドレス "13X.XXX.2.110") 3 2 0 6 への接続情報が格納されている。この接続情報によって、SF 3 2 0 3 からTerm3 3 2 0 6 への接続ポートがポート3であることを検出可能である。また、IF 3 2 0 2 からR 3 2 0 1 への接続ポートとIF 3 2 0 2 からTerm2 3 2 0 5 への接続ポートが異なることから、IF 3 2 0 2

はR3 2 0 1とTerm23 2 0 5の中間に接続している機器であることが検出可能である。同様に、IF3 2 0 2からR3 2 0 1への接続ポートとIF3 2 0 2からTerm3 2 0 6への接続ポートが異なることから、IF3 2 0 2はR3 2 0 1とTerm3 2 0 6の中間に接続している機器であることが検出可能である。

【 0 0 5 6 】

また、SF3 2 0 3からTerm23 2 0 5への接続ポートとSF3 2 0 3からTerm3 2 0 6への接続ポートが異なることから、SF3 2 0 3はTerm23 2 0 5とTerm3 2 0 6の中間に接続している機器であることが検出可能である。従って、SF3 2 0 3はIF3 2 0 2とTerm23 2 0 5、Term3 2 0 6の中間に接続している機器であることが検出可能であり、IF3 2 0 2はSF3 2 0 3の親であることが検出可能である。さらに、IF3 2 0 2からSF3 2 0 3への接続ポートとIF3 2 0 2からTerm23 2 0 5、Term3 2 0 6への接続ポートは等しいため、IF3 2 0 2からSF3 2 0 3への接続ポートはポート1であることが検出可能である。

また、IF3 2 0 2からR3 2 0 1への接続ポートとIF3 2 0 2からTerm13 2 0 4への接続ポートが異なることから、IF3 2 0 2はR3 2 0 1とTerm13 2 0 4の中間に接続している機器であることが検出可能である。

【 0 0 5 7 】

同様に、IF3 2 0 2からSF3 2 0 3への接続ポートとIF3 2 0 2からTerm13 2 0 4への接続ポートが異なることから、IF3 2 0 2はSF3 2 0 3とTerm13 2 0 4の中間に接続している機器であることが検出可能である。従って、SF3 2 0 3からIF3 2 0 2への接続ポートとSF3 2 0 3からTerm13 2 0 4への接続ポートは等しい。従って、SF3 2 0 3からIF3 2 0 2への接続ポートはポート1であることが検出可能である。

このようにして、R-IF-SFモデルでは、IF3 2 0 2とTerm13 2 0 4～Term3 2 0 6の接続情報とSF3 2 0 3とTerm13 2 0 4～Term3 2 0 6の接続情報がP Fテーブル1 1 2 4に格納されているという条件下で機器の接続ポートと親子関係の検出が可能である。

【 0 0 5 8 】

図3 4は、本実施形態のR-SF-CFモデルの接続検出の仕組みを示す図であり、R-S

F-CFモデルの一例として、R(I Pアドレス "13X.XXX.2.1") 3 4 0 1のポート2とNF(I Pアドレスなし) 3 4 0 2のポート3に接続関係があり、NF(I Pアドレスなし)のポート2とSF(I Pアドレス "13X.XXX.2.246") 3 4 0 3のポート2に接続関係があり、さらにSF(I Pアドレス "13X.XXX.2.246") 3 4 0 3のポート1とCF(I Pアドレス "13X.XXX.2.243") 3 4 0 4のポート1に接続関係があり、NF(I Pアドレスなし) 3 4 0 2のポート1の先に任意のTerm1(I Pアドレス "13X.XXX.2.51") 3 2 0 5が接続されており、CF(I Pアドレス "13X.XXX.2.243") 3 4 0 4のポート2の先に任意のTerm2(I Pアドレス "13X.XXX.2.2") 3 4 0 6が接続されている場合を示している。

【 0 0 5 9 】

図 3 5 は本実施形態のR-SF-CFモデルの接続検出に利用する P F テーブル 1 1 2 4 のエントリ例を示すものであり、図 3 4 に示したR-SF-CFモデルに対する P F テーブル 1 1 2 4 のエントリを例示している。

エントリ 3 5 0 1 にSF(I Pアドレス "13X.XXX.2.246") 3 4 0 3 からTerm2(I Pアドレス "13X.XXX.2.2") 3 4 0 6 への接続情報が格納されている。この接続情報によって、SF 3 4 0 3 からTerm2 3 4 0 6 への接続ポートがポート1であることを検出可能である。

また、エントリ 3 5 0 2 にSF(I Pアドレス "13X.XXX.2.246") 3 4 0 3 からTerm1(I Pアドレス "13X.XXX.2.51") 3 4 0 5 への接続情報が格納されている。この接続情報によって、SF 3 4 0 3 からTerm1 3 4 0 5 への接続ポートがポート2であることを検出可能である。

また、エントリ 3 5 0 3 にCF(I Pアドレス "13X.XXX.2.243") 3 4 0 4 からR(I Pアドレス "13X.XXX.2.1") 3 4 0 1 への接続情報が格納されている。この接続情報によって、CF 3 4 0 4 からR 3 4 0 1 への接続ポートがポート1であることを検出可能である。

【 0 0 6 0 】

また、エントリ 3 5 0 4 にCF(I Pアドレス "13X.XXX.2.243") 3 3 0 4 からTerm1(I Pアドレス "13X.XXX.2.51") 3 4 0 5 への接続情報が格納されている。この接続情報によって、CF 3 4 0 4 からTerm1 3 4 0 5 への接続ポートがポート1で

あることを検出可能である。

また、エントリ 3 5 0 5 に CF (IP アドレス "13X.XXX.2.243") 3 4 0 4 から SF (IP アドレス "13X.XXX.2.246") 3 4 0 3 への接続情報が格納されている。この接続情報によって、CF 3 4 0 4 から SF 3 4 0 3 への接続ポートがポート 1 であることを検出可能である。

また、エントリ 3 5 0 6 に CF (IP アドレス "13X.XXX.2.243") 3 4 0 4 から Term2 (IP アドレス "13X.XXX.2.2") 3 4 0 6 への接続情報が格納されている。この接続情報によって、CF 3 4 0 4 から Term2 3 4 0 6 への接続ポートがポート 2 であることを検出可能である。そして、CF 3 4 0 4 から R 3 4 0 1 への接続ポートと CF 3 4 0 4 から SF 3 4 0 3 への接続ポートが等しいことから、SF 3 4 0 3 は CF 3 4 0 4 の親または兄弟であり、CF 3 4 0 4 から SF 3 4 0 3 への接続ポートは 1 であることが検出可能である。また、SF 3 4 0 3 から Term1 3 4 0 5 への接続ポートと SF 3 4 0 3 から Term2 3 4 0 6 への接続ポートが異なることから、SF 3 4 0 3 は Term1 3 4 0 5 と Term2 3 4 0 6 の間に接続している機器であることが検出可能である。

【 0 0 6 1 】

また、CF 3 4 0 4 から SF 3 4 0 3 への接続ポートと CF 3 4 0 4 から Term2 3 4 0 6 への接続ポートが異なることから、CF 3 4 0 4 は SF 3 4 0 3 と Term2 3 4 0 6 の間に接続している機器であり、SF 3 4 0 3 から Term2 3 4 0 6 への接続ポートと SF 3 4 0 3 から CF 3 4 0 4 への接続ポートは等しい。従って、SF 3 4 0 3 から CF 3 4 0 4 への接続ポートは 1 であることが検出可能である。また、SF 3 4 0 3 から R 3 4 0 1 への接続ポートの検出が不可能であるため、SF 3 4 0 3 と CF 3 4 0 4 の親子関係は検出不能である (SF 3 4 0 3 と CF 3 4 0 4 の間に NF 3 4 0 2 が接続している場合は SF 3 4 0 3 と CF 3 4 0 4 は兄弟になる)。

このようにして、R-SF-CF モデルでは、SF 3 4 0 3 と Term1 3 4 0 5、Term2 3 4 0 6 の接続情報と CF と Term1 3 4 0 5、Term2 3 4 0 6 の接続情報が P F テーブル 1 1 2 4 に格納されているという条件下で機器の接続ポートだけの検出が可能になる。

【 0 0 6 2 】

図 3 6 は、本実施形態の R-SF-IF モデルの接続検出の仕組みを示す図であり、R-SF-IF モデルの一例として、R (IP アドレス "13X.XXX.2.1") 3 6 0 1 のポート 2 と NF (IP アドレスなし) 3 6 0 2 のポート 3 に接続関係があり、NF (IP アドレスなし) 3 6 0 2 のポート 2 と SF (IP アドレス "13X.XXX.2.246") 3 6 0 3 のポート 2 に接続関係があり、さらに SF (IP アドレス "13X.XXX.2.246") 3 6 0 3 のポート 1 と IF (IP アドレス "13X.XXX.2.243") 3 6 0 4 のポート 1 に接続関係があり、NF (IP アドレスなし) 3 6 0 2 のポート 1 の先に任意の Term1 (IP アドレス "13X.XXX.2.51") 3 6 0 5 が接続されており、さらに SF (IP アドレス "13X.XXX.2.246") 3 6 0 3 のポート 3 の先に任意の Term2 (IP アドレス "13X.XXX.2.102") 3 6 0 6 が接続されており、IF (IP アドレス "13X.XXX.2.243") 3 6 0 4 のポート 2 の先に任意の Term3 (IP アドレス "13X.XXX.2.2") 3 6 0 7 が接続されている場合を示している。

【 0 0 6 3 】

図 3 7 は本実施形態の R-SF-IF モデルの接続検出に利用する P F テーブル 1 1 2 4 のエントリ例を示すものであり、図 3 6 に示した R-SF-IF モデルに対する P F テーブル 1 1 2 4 のエントリを例示している。

エントリ 3 7 0 1 に SF (IP アドレス "13X.XXX.2.246") 3 6 0 3 から Term3 (IP アドレス "13X.XXX.2.2") 3 6 0 7 への接続情報が格納されている。この接続情報によって、SF 3 6 0 3 から Term3 3 6 0 7 への接続ポートがポート 1 であることを検出可能である。

また、エントリ 3 7 0 2 に SF (IP アドレス "13X.XXX.2.246") 3 6 0 3 から Term1 (IP アドレス "13X.XXX.2.51") 3 6 0 5 への接続情報が格納されている。この接続情報によって、SF 3 6 0 3 から Term1 3 6 0 5 への接続ポートがポート 2 であることを検出可能である。

また、エントリ 3 7 0 3 に SF (IP アドレス "13X.XXX.2.246") 3 6 0 3 から Term2 (IP アドレス "13X.XXX.2.102") 3 6 0 6 への接続情報が格納されている。この接続情報によって、SF 3 6 0 3 から Term2 3 6 0 6 への接続ポートがポート 3 であることを検出可能である。

【 0 0 6 4 】

また、エントリ 3 7 0 4 に IF (I P アドレス "13X.XXX.2.243") 3 6 0 4 から R (I P アドレス "13X.XXX.2.1") 3 6 0 1 への接続情報が格納されている。この接続情報によって、IF 3 6 0 4 から R 3 6 0 1 への接続ポートがポート 1 であることを検出可能である。

また、エントリ 3 7 0 5 に IF (I P アドレス "13X.XXX.2.243") 3 6 0 4 から Term1 (I P アドレス "13X.XXX.2.51") 3 6 0 5 への接続情報が格納されている。この接続情報によって、IF 3 6 0 4 から Term1 3 6 0 5 への接続ポートがポート 1 であることを検出可能である。

また、エントリ 3 7 0 6 に IF (I P アドレス "13X.XXX.2.243") 3 6 0 4 から Term2 (I P アドレス "13X.XXX.2.102") 3 6 0 6 への接続情報が格納されている。この接続情報によって、IF 3 6 0 4 から Term2 3 6 0 6 への接続ポートがポート 1 であることを検出可能である。

また、エントリ 3 7 0 7 に IF (I P アドレス "13X.XXX.2.243") 3 6 0 4 から Term3 (I P アドレス "13X.XXX.2.2") 3 6 0 7 への接続情報が格納されている。この接続情報によって、IF 3 6 0 4 から Term3 3 6 0 7 への接続ポートがポート 2 であることを検出可能である。また、IF 3 6 0 4 から R 3 6 0 1 への接続ポートと IF 3 6 0 4 から Term1 3 6 0 5 への接続ポートが等しいことから、Term1 3 6 0 5 は R 3 6 0 1 と IF 3 6 0 4 の中間に接続している機器であることが検出可能である。

【 0 0 6 5 】

同様に、IF 3 6 0 4 から R 3 6 0 1 への接続ポートと IF 3 6 0 4 から Term2 3 6 0 6 への接続ポートが等しいことから、Term2 3 6 0 6 は R 3 6 0 1 と IF 3 6 0 4 の中間に接続している機器であることが検出可能である。また、SF 3 6 0 3 から Term1 3 6 0 5 への接続ポートと SF 3 6 0 3 から Term2 3 6 0 6 への接続ポートが異なることから、SF 3 6 0 3 は Term1 3 6 0 5 と Term2 3 6 0 6 の中間に接続している機器であることが検出可能である。

従って、IF 3 6 0 4 から SF 3 6 0 3 への接続ポートと IF 3 6 0 4 から Term1 3 6 0 5、Term2 3 6 0 6 への接続ポートは等しいことから、IF 3 6 0 4 から SF 3 6 0 3 への接続ポートは 1 であることが検出可能である。また、IF 3 6 0 4 から R

3 6 0 1 への接続ポートと IF 3 6 0 4 から Term 3 3 6 0 7 への接続ポートが異なることから、IF 3 6 0 4 は R 3 6 0 1 と Term 3 3 6 0 7 の間に接続している機器であることが検出可能である。さらに、SF 3 6 0 3 は R 3 6 0 1 と IF 3 6 0 4 の間に接続していることから、IF 3 6 0 4 は SF 3 6 0 3 と Term 3 3 6 0 7 の間に接続していることが検出可能である。

【 0 0 6 6 】

従って、SF 3 6 0 3 から IF 3 6 0 4 への接続ポートは SF 3 6 0 3 から Term 3 3 6 0 7 の接続ポートに等しいため、SF 3 6 0 3 から IF 3 6 0 4 への接続ポートは 1 であることが検出可能である。また、SF 3 6 0 3 から R 3 6 0 1 への接続ポートの検出が不可能であるため、SF 3 6 0 3 と IF 3 6 0 4 の親子関係は検出不能である(図 3 7 は兄弟の例を示す)。

このようにして、R-IF-SFモデルでは、SF 3 6 0 3 と Term 1 3 6 0 5 ~ Term 3 3 6 0 7 の接続情報と IF 3 6 0 4 と Term 1 3 6 0 5 ~ Term 3 3 6 0 7 の接続情報が P F テーブル 1 1 2 4 に格納されているという条件下で機器の接続ポートだけの検出が可能になる。

【 0 0 6 7 】

図 3 8 は本実施形態の R-SF-SFモデルの接続検出の仕組みを示す図であり、R-SF-SFモデルの一例として、(IP アドレス "13X.XXX.2.1") 3 8 0 1 のポート 2 と NF (IP アドレスなし) 3 8 0 2 のポート 3 に接続関係があり、NF (IP アドレスなし) 3 8 0 2 のポート 2 と SF 1 (IP アドレス "13X.XXX.2.246") 3 8 0 3 のポート 2 に接続関係があり、さらに SF 1 (IP アドレス "13X.XXX.2.246") 3 8 0 3 のポート 1 と SF 2 (IP アドレス "13X.XXX.2.243") 3 8 0 4 のポート 1 に接続関係があり、NF (IP アドレスなし) 3 8 0 2 のポート 1 の先に任意の Term 1 (IP アドレス "13X.XXX.2.51") 3 8 0 5 が接続されており、SF 2 (IP アドレス "13X.XXX.2.243") 3 8 0 4 のポート 2 の先に任意の Term 2 (IP アドレス "13X.XXX.2.2") 3 8 0 6 が接続されている場合を示している。

【 0 0 6 8 】

図 3 9 は本実施形態の R-SF-SFモデルの接続検出に利用する P F テーブル 1 1 2 4 のエントリ例を示すものであり、図 3 8 に示した R-SF-SFモデルに対する P

Fテーブル1124のエントリを例示している。

まず、エントリ3901にSF1(IPアドレス "13X.XXX.2.246") 3803からTerm2(IPアドレス "13X.XXX.2.2") 3806への接続情報が格納されている。この接続情報によって、SF13803からTerm23806への接続ポートがポート1であることを検出可能である。

また、エントリ3902にSF1(IPアドレス "13X.XXX.2.246") 3803からTerm1(IPアドレス "13X.XXX.2.51") 3805への接続情報が格納されている。この接続情報によって、SF13803からTerm13805への接続ポートがポート2であることを検出可能である。

【0069】

エントリ3903にSF2(IPアドレス "13X.XXX.2.246") 3804からTerm1(IPアドレス "13X.XXX.2.51") 3805への接続情報が格納されている。この接続情報によって、SF23804からTerm13805への接続ポートがポート1であることを検出可能である。

エントリ3903にSF2(IPアドレス "13X.XXX.2.243") 3804からTerm2(IPアドレス "13X.XXX.2.2") 3806への接続情報が格納されている。この接続情報によって、SF23804からTerm23806への接続ポートがポート2であることを検出可能である。

しかし、PFテーブル1124のエントリからは、SF13803のポート1とSF23804のポート1に接続関係がある場合とSF13803のポート2とSF23804のポート2に接続関係がある場合を判別できないため、接続ポートは検出できない。また、SF13803とR3801の接続関係やSF23804とR3801の接続関係を検出不能なことにより親子関係の検出も不能である。R-SF-SFモデルは、任意の条件下で機器の接続ポートと親子関係の検出が不可能である。

【0070】

図40は本実施形態のR-CFモデルの接続検出の仕組みを示す図であり、R-CFモデルの一例として、R(IPアドレス "13X.XXX.2.1") 4001のポート2とCF(IPアドレス "13X.XXX.2.246") 4002のポート2に接続関係がある場合を示している。

【 0 0 7 1 】

図 4 1 は本実施形態の R-CF モデルの接続検出に利用する P F テーブル 1 1 2 4 のエントリ例を示すものであり、図 4 0 に示した R-CF モデルに対する P F テーブル 1 1 2 4 のエントリを例示している。

エントリ 4 1 0 1 に R (I P アドレス "13X.XXX.2.1") 4 0 0 1 から CF (I P アドレス "13X.XXX.2.246") 4 0 0 2 への接続情報が格納されている。この接続情報によって、R 4 0 0 1 から CF 4 0 0 2 への接続ポートがポート 2 であることを検出可能である。

また、CF (I P アドレス "13X.XXX.2.246") 4 0 0 2 から R (I P アドレス "13X.XXX.2.1") 4 0 0 1 への接続情報が格納されている。この接続情報によって、CF 4 0 0 2 から R 4 0 0 1 への接続ポートがポート 2 であることを検出可能である。

なお、R (I P アドレス "13X.XXX.2.1") 4 0 0 1 から CF (I P アドレス "13X.XXX.2.246") 4 0 0 2 への接続情報がない場合でも、R (I P アドレス "13X.XXX.2.1") 4 0 0 1 と "13X.XXX.2.*" ネットワーク上の任意の機器の接続情報が存在すれば、そのポートが R 4 0 0 1 から CF 4 0 0 2 への接続ポートになる。また、CF (I P アドレス "13X.XXX.2.246") 4 0 0 2 から R (I P アドレス "13X.XXX.2.1") 4 0 0 1 への接続情報がない場合でも、CF (I P アドレス "13X.XXX.2.246") 4 0 0 2 と別セグメントに接続している機器の接続情報が存在すれば、そのポートが CF 4 0 0 2 から R 4 0 0 1 への接続ポートになる。

このようにして、R-CF モデルでは、任意の条件下で機器の接続ポートと親子関係の検出が可能である。

【 0 0 7 2 】

図 4 2 は本実施形態の R-IF モデルの接続検出の仕組みを示す図であり、R-IF モデルの一例として、R (I P アドレス "13X.XXX.2.1") 4 2 0 1 のポート 2 と IF (I P アドレス "13X.XXX.2.246") 4 2 0 2 のポート 2 に接続関係がある場合を示している。

【 0 0 7 3 】

図 4 3 は本実施形態の R-IF モデルの接続検出に利用する P F テーブル 1 1 2 4 のエントリ例を示すものであり、図 4 2 に示した R-IF モデルに対する P F テーブル

ル 1 1 2 4 のエントリを例示している。

エントリ 4 3 0 1 に R (I P アドレス "13X.XXX.2.1") 4 2 0 1 から I F (I P アドレス "13X.XXX.2.246") 4 2 0 2 への接続情報が格納されている。この接続情報によって、R 4 2 0 1 から I F 4 2 0 2 への接続ポートがポート 2 であることを検出可能である。

また、エントリ 4 3 0 2 に I F (I P アドレス "13X.XXX.2.246") 4 2 0 2 から R (I P アドレス "13X.XXX.2.1") 4 2 0 1 への接続情報が格納されている。この接続情報によって、I F 4 2 0 2 から R 4 2 0 1 への接続ポートがポート 2 であることを検出可能である。

なお、R (I P アドレス "13X.XXX.2.1") 4 2 0 1 から I F (I P アドレス "13X.XXX.2.246") 4 2 0 2 への接続情報がない場合でも、R (I P アドレス "13X.XXX.2.1") 4 2 0 1 と "13X.XXX.2.*" ネットワーク上の任意の機器の接続情報が存在すれば、そのポートが R 4 2 0 1 から I F 4 2 0 2 への接続ポートになる。また、I F (I P アドレス "13X.XXX.2.246") 4 2 0 2 から R (I P アドレス "13X.XXX.2.1") 4 2 0 1 への接続情報がない場合でも、I F (I P アドレス "13X.XXX.2.246") 4 2 0 2 と別セグメントに接続している機器の接続情報が存在すれば、そのポートが I F 4 2 0 2 から R 4 2 0 1 への接続ポートになる。

このようにして、R-IF モデルでは、任意の条件下で機器の接続ポートと親子関係の検出が可能である。

【 0 0 7 4 】

図 4 4 は本実施形態の R-SF モデルの接続検出の仕組みを示す図であり、R-SF モデルの一例として、R (I P アドレス "13X.XXX.2.1") 4 4 0 1 のポート 2 と S F (I P アドレス "13X.XXX.2.246") 4 4 0 2 のポート 2 に接続関係があり、R (I P アドレス "13X.XXX.2.1") 4 4 0 1 のポート 1 の先には任意の Term1 (I P アドレス "13X.XXX.1.1") 4 4 0 3 が接続している場合を示している。

【 0 0 7 5 】

図 4 5 は本実施形態の R-SF モデルの接続検出に利用する P F テーブル 1 1 2 4 のエントリ例を示すものであり、図 4 4 に示した R-SF モデルに対する P F テーブル 1 1 2 4 のエントリを例示している。

エントリ 4 5 0 1 に R (I P アドレス "13X.XXX.2.1") 4 4 0 1 から SF (I P アドレス "13X.XXX.2.246") 4 4 0 2 への接続情報が格納されている。この接続情報によって、R 4 4 0 1 から SF 4 4 0 2 への接続ポートがポート 2 であることを検出可能である。

また、エントリ 4 5 0 2 に SF (I P アドレス "13X.XXX.2.246") から別セグメントに接続している機器の接続情報 ("13X.XXX.2.*" のネットワークの機器ではない機器 Term1 ("13X.XXX.1.1") 4 4 0 3 が格納されている。この接続情報によって、SF 4 4 0 2 から R 4 4 0 1 への接続ポートがポート 2 であることを検出可能である。

なお、R (I P アドレス "13X.XXX.2.1") 4 4 0 1 から SF (I P アドレス "13X.XXX.2.246") 4 4 0 2 への接続情報がない場合でも、R (I P アドレス "13X.XXX.2.1") 4 4 0 1 と "13X.XXX.2.*" ネットワーク上の任意の機器の接続情報が存在すれば、そのポートが R 4 4 0 1 から SF 4 4 0 2 への接続ポートになる。

このようにして、R-SF モデルでは、別セグメントに接続している機器の接続情報が取得可能であるという条件下で機器の接続ポートと親子関係の検出が可能である。

【 0 0 7 6 】

図 4 6 および図 4 7 は、本実施形態のネットワーク中継装置同士の接続検出方法を説明する図である。

図 4 6 及び図 4 7 は、図 2 2 ～図 4 5 で示したネットワーク中継装置同士の接続関係や親子関係の検出条件を表形式でまとめたものである。

接続モデル 4 6 0 1、4 7 0 1 ごとに検出条件を設定しており、親から子への接続ポート 4 6 0 2、4 7 0 2、子から親への接続ポート 4 6 0 3、4 7 0 3、親子関係 4 6 0 4、4 7 0 4 の検出可能性を示している。

「○」印の項目は接続検出するための条件 4 6 0 5、4 7 0 5 に関係なく検出が可能であることを示しており、「△」印の項目は接続検出するための条件を満たしている場合に限り検出が可能であることを示しており、「×」印の項目は任意の条件下で検出が不可能であることを示している。

【 0 0 7 7 】

図 4 8 は、本実施形態のCF-Termモデルの接続検出の仕組みを示す図であり、CF-Termモデルの一例として、CF(I P アドレス "13X.XXX.2.246") 4 8 0 1 のポート 3 と Term1(I P アドレス "13X.XXX.2.102") 4 8 0 2 に接続関係がある場合を示している。

【 0 0 7 8 】

図 4 9 は本実施形態のCF-Termモデルの接続検出に利用する P F テーブル 1 1 2 4 のエントリ例を示す図であり、図 4 8 に示したCF-Termモデルに対する P F テーブルのエントリを例示している。

エントリ 4 9 0 1 にCF(I P アドレス "13X.XXX.2.246") 4 8 0 1 からTerm1(I P アドレス "13X.XXX.2.102") 4 8 0 2 への接続情報が格納されている。この接続情報によって、CF 4 8 0 1 からTerm1 4 8 0 2 への接続ポートがポート 3 であることを検出可能である。

この場合、CF 4 8 0 1 のポート 3 の先に任意の台数の機器が接続している場合でも、P F テーブル 1 1 2 4 には機器の台数分の接続情報が格納されるため、任意の台数のTermの検出が可能である。

このようにして、CF-Termモデルでは、任意の条件下で機器の接続ポートと親子関係の検出が可能である。

図 5 0 は、本実施形態のIF-Termモデルの接続検出の仕組みを示す図であり、IF-Termモデルの一例として、IF(I P アドレス "13X.XXX.2.246") 5 0 0 1 のポート 3 と Term1(I P アドレス "13X.XXX.2.102") 5 0 0 2 に接続関係がある場合を示している。

【 0 0 7 9 】

図 5 1 は、本実施形態のIF-Termモデルの接続検出に利用する P F テーブル 1 1 2 4 のエントリ例を示す図であり、図 5 0 に示したIF-Termモデルに対する P F テーブル 1 1 2 4 のエントリを例示している。

エントリ 5 1 0 1 にIF(I P アドレス "13X.XXX.2.246") 5 0 0 1 からTerm1(I P アドレス "13X.XXX.2.102") 5 0 0 2 への接続情報が格納されている。この接続情報によって、IF 5 0 0 1 からTerm1 5 0 0 2 への接続ポートがポート 3 であることを検出可能である。

この場合、IF 5 0 0 1 のポート3の先に任意の台数の機器が接続している場合でも、P F テーブル 1 1 2 4 には機器の台数分の接続情報が格納されるため、任意の台数のTermの検出が可能である。

このようにして、IF-Termモデルでは、任意の条件下で機器の接続ポートと親子関係の検出が可能である。

【 0 0 8 0 】

図 5 2 は本実施形態のSF-Termモデルの接続検出の仕組みを示す図であり、SF-Termモデルの一例として、SF(I P アドレス "13X.XXX.2.246") 5 2 0 1 のポート3とTerm1(I P アドレス "13X.XXX.2.102") 5 2 0 2 に接続関係がある場合を示している。

図 5 3 は本実施形態のIF-Termモデルの接続検出に利用するP F テーブル 1 1 2 4 のエントリ例を示す図であり、図 5 2 に示したSF-Termモデルに対するP F テーブル 1 1 2 4 のエントリを例示している。

エントリ 5 3 0 1 にSF(I P アドレス "13X.XXX.2.246") 5 2 0 1 からTerm1(I P アドレス "13X.XXX.2.102") 5 2 0 2 への接続情報が格納されている。この接続情報によって、SFからTerm1への接続ポートがポート3であることを検出可能である。

この場合、SF 5 2 0 1 のポート3の先に複数台の機器が接続している場合は、P F テーブル 1 1 2 4 には機器 1 台数分の接続情報が格納されるため、任意の 1 台のTermの検出が可能である。

このようにして、SF-Termモデルでは、ネットワーク中継装置の各ポートに 1 台の機器が接続しているという条件下で機器の接続ポートと親子関係の検出が可能である。

【 0 0 8 1 】

図 5 4 は本実施形態のネットワーク構成自動認識方法におけるネットワーク中継装置と端末装置の接続検出方法を説明する図である。図 5 4 は図 4 8 ～図 5 3 で示したネットワーク中継装置と端末装置の接続関係や親子関係の検出条件を表形式でまとめたものである。

接続モデル 5 4 0 1 ごとに端末装置の接続検出 5 3 0 2 の可能性を示しており

、接続検出するための条件 5 4 0 3 によって接続検出可能性が変化する。

「○」印の項目は接続検出するための条件に関係なく検出が可能であることを示しており、「△」印の項目は接続検出するための条件を満たしている場合に限り検出が可能であることを示している。

【 0 0 8 2 】

図 5 5 は本実施形態の複数のモデルを組合わせることによる親子関係の検出を説明する図である。図 5 5 は R-CF-CF モデルと R-CF-SF モデルを組合わせ、R-SF-CF モデルの親子関係を検出する場合の一例を示している。

ここでは、R (IP アドレス "13X.XXX.2.1") 5 5 0 1 のポート 2 と CF1 (IP アドレス "13X.XXX.2.246") 5 5 0 2 のポート 2 に接続関係があり、さらに CF1 (IP アドレス "13X.XXX.2.246") 5 5 0 2 のポート 1 と SF (IP アドレス "13X.XXX.2.243") 5 5 0 3 のポート 1 に接続関係があり、SF (IP アドレス "13X.XXX.2.243") 5 5 0 3 のポート 2 と CF2 (IP アドレス "13X.XXX.2.247") 5 5 0 4 のポート 2 に接続関係があり、さらに CF1 (IP アドレス "13X.XXX.2.246") 5 5 0 2 のポート 3 の先に任意の Term1 (IP アドレス "13X.XXX.2.102") 5 5 0 5 が接続されており、さらに CF2 (IP アドレス "13X.XXX.2.247") 5 5 0 4 のポート 1 の先に任意の Term2 (IP アドレス "13X.XXX.2.2") 5 5 0 6 が接続されている場合を示している。

【 0 0 8 3 】

図 5 6 は本実施形態の複数のモデルを組合わせることによる親子関係の検出に利用する TS テーブル 1 1 2 5 のエントリを示す図であり、図 5 5 に示した R-CF-CF モデルと R-CF-SF モデルから R-SF-CF モデルを検出するためのエントリを例示している。

R-CF-SF モデルでは、Term1 への接続情報を CF と SF の両方が保持している条件下で親子関係の検出が可能であるため、CF1 (IP アドレス "13X.XXX.2.246") 5 5 0 2 は SF (IP アドレス "13X.XXX.2.243") 5 5 0 3 の親であることを示すエントリ 5 6 0 1 が TS テーブル 1 1 2 5 に格納される。

また、R-CF-CF モデルでは、任意の条件下で親子関係の検出が可能であるため、CF1 (IP アドレス "13X.XXX.2.246") 5 5 0 2 は CF2 (IP アドレス "13X.XXX.2.247") 5 5 0 4 の親であることを示すエントリ 5 6 0 2 が TS テーブル 1 1 2 5

に格納される。なお、R-SF-CFモデルでは親子関係の検出が不可能であるため、SF(IPアドレス "13X.XXX.2.243") 5 5 0 3とCF2(IPアドレス "13X.XXX.2.247") 5 5 0 4の親子関係は不明である。

【 0 0 8 4 】

従って、図 5 6 では接続関係(接続ポート)は検出可能であるが、親子関係の検出は不可能である場合を示す例として、CF2(IPアドレス "13X.XXX.2.247") 5 5 0 4はSF(IPアドレス "13X.XXX.2.243") 5 5 0 3の親であることを示すエントリ 5 6 0 3とSF(IPアドレス "13X.XXX.2.243") 5 5 0 3はCF2(IPアドレス "13X.XXX.2.247") 5 5 0 4の親であることを示すエントリ 5 6 0 4を同時に格納している。SF 5 5 0 3からCF1 5 5 0 2への接続ポートとSF 5 5 0 3からCF2 5 5 0 4への接続ポートが異なることから、SF 5 5 0 3はCF1 5 5 2とCF2 5 5 0 4の中間に接続していることが検出可能である。また、CF1 5 5 0 2はSF 5 5 0 3の親であることから、SF 5 5 0 3はCF2 5 5 0 4の親であることが検出可能である。

このようにして、R-SF-CFモデルは親子関係の検出が不可能であるが、R-CF-CFモデルとR-CF-SFモデルから検出可能な親子関係を組合わせることで、親子関係の検出が可能である。

【 0 0 8 5 】

図 5 7 は本実施形態のNon Intelligent Hubの接続の予測方法を説明する図であり、ここでは、Non Intelligent Hubの接続の予測する一例として、Unit(IPアドレス "13X.XXX.2.246") 5 7 0 1のポート1とNF(IPアドレスなし) 5 7 0 2のポート1に接続関係があり、NF(IPアドレスなし) 5 7 0 2のポート2の先に任意のTerm1(IPアドレス "13X.XXX.2.98") 5 7 0 3が接続されており、NF(IPアドレスなし) 5 7 0 2のポート3の先に任意のTerm2(IPアドレス "13X.XXX.2.13") 5 7 0 4が接続されている場合を示している。

【 0 0 8 6 】

図 5 8 は、本実施形態のNon Intelligent Hubの接続の予測に利用するTSテーブル 1 1 2 5のエントリ例を示す図であり、図 5 7 に示したNon Intelligent Hubの接続の予測に対するTSテーブル 1 1 2 5のエントリを例示している。

エントリ 5 8 0 1 には Unit (I P アドレス "13X.XXX.2.246") 5 7 0 1 のポート 1 の先に Term1 (I P アドレス "13X.XXX.2.98") 5 7 0 3 が子として接続されていることを示す接続情報が格納されている。

【 0 0 8 7 】

同様に、エントリ 5 8 0 2 には、Unit (I P アドレス "13X.XXX.2.246") 5 7 0 1 のポート 1 の先に Term2 (I P アドレス "13X.XXX.2.13") 5 7 0 4 が子として接続されていることを示す接続情報が格納されている。

これによって、ネットワーク中継装置の共通のポートに複数の機器が子として接続しており、別のネットワーク中継装置が存在しない場合は Non Intelligent Hub が少なくとも 1 台以上接続していることが検出可能である。そして、ネットワーク中継装置の共通のポートに複数の機器が子として接続しており、別のネットワーク中継装置が存在する場合でもネットワーク中継装置とネットワーク中継装置の接続ポートの間に複数の機器が接続しており、接続機器の中にネットワーク中継装置を含まない場合は Non Intelligent Hub が少なくとも 1 台以上接続していることが検出可能である。

【 0 0 8 8 】

図 5 9 は本実施形態の非稼動中端末装置の検出方法を説明する図であり、R (I P アドレス "13X.XXX.2.1") 5 9 0 1 のポート 2 と Unit (I P アドレス "13X.XXX.2.243") 5 9 0 2 のポート 2 に接続関係があり、Unit (I P アドレス ("13X.XXX.2.243") 5 9 0 2 のポート 1 の先に先に任意の Term (I P アドレス "13X.XXX.2.2") 5 9 0 3 が接続されているが、Term 5 9 0 3 は非稼動中端末装置である場合を示している。

図 5 9 の例では、非稼動中端末装置 5 9 0 3 の接続関係や親子関係を検出する一例として、ネットワーク中の I P アドレスに対してポーリングを行い、ポーリングに応答がない I P アドレスの機器が存在した場合、その I P アドレスに対応する機器が存在しない、または非稼動中であるとし、T I テーブル 1 1 2 3 (図 1 4) の alive 値が FALSE のエントリを追加する。次に、Router の ARP キャッシュを参照し、ポーリングに応答を返さない I P アドレスのエントリが ARP キャッシュに含まれている場合には、その I P アドレスに対応する機器は非稼動中である

ものとして検出する。また、ネットワーク中継装置同士の接続関係や親子関係の検出に利用するMIBオブジェクトに非稼動中端末装置の接続情報が含まれている場合には、PFテーブル1124やTSテーブル1125に非稼動中端末装置のエントリの作成が可能になるため、非稼動中端末装置の接続関係や親子関係の検出が可能になる。

【0089】

図60は本実施形態の接続先の変更の検出方法を説明する図であり、Unit(IPアドレス "13X.XXX.2.243") 6001のポート2の先に任意のTerm(IPアドレス "13X.XXX.2.2") 6002が接続されていたが、Unit 6001のポート3に接続先を変更した例を示している。

【0090】

図61は本実施形態の接続先の変更の検出に利用するTSテーブル1125のエントリ例を示す図であり、図60に示した接続先の変更の検出に対するエントリを例示している。

接続先変更前のTSテーブル1125のエントリ6101には、Unit(IPアドレス "13X.XXX.2.243") 6001のポート2の先にはTerm(IPアドレス "13X.XXX.2.2") 6002が子として接続されていることを示す接続情報が格納されている。接続先変更後のTSテーブルのエントリでは、Unit(IPアドレス "13X.XXX.2.243") 6001のポート2の先にTerm(IPアドレス "13X.XXX.2.2") 6002が子として接続しているエントリ6102とUnit(IPアドレス "13X.XXX.2.243") 6001のポート3の先にTerm(IPアドレス "13X.XXX.2.2") 6002が子として接続しているエントリ6103が存在している。接続先の変更前のTSテーブル1125と接続先の変更後のTSテーブル1125では、機器の接続先の情報も変更されるため、定期的にTSテーブル1125を作成し、差分をとることで、接続先の変更の検出が可能になる。

また、エントリ6102のように古い接続情報のキャッシュがMIBオブジェクトが残る場合でも問題ない。また、機器のIPアドレスを変更した場合には、IPアドレスの機器がネットワーク上に新たに追加されたという形で検出可能である。

【 0 0 9 1 】

図 6 2 は本実施形態の図面表示プログラム 1 1 0 4 が作成するネットワーク構成図面の表示例を示す図である。

図面表示プログラム 1 1 0 4 の G U I は、Network Map 表示部分 6 2 0 1、Terminal Information 表示部分 6 2 0 2 から構成される。

Network Map 表示部分 6 2 0 1 には、オートディスカバリモジュールを実行することにより自動検出したネットワークセグメントの構成を図示のように木構造表示する。この Network Map 表示部分 6 2 0 1 の任意の機器表示にマウス等のポインティングデバイスを利用してカーソルを充てた場合、該機器表示を符号 6 2 0 3 で示すように反転表示し、Terminal Information 表示部分 6 2 0 2 に機器の情報を表示する。

【 0 0 9 2 】

図 6 2 の例では、T I テーブル 1 1 2 3 (図 1 4) の該当する機器の情報を表示した例を示している。また、接続されているものと予測した Non Intelligent Hub 6 2 0 4 を表示することも可能である。

ユーザは Terminal Information 表示部分 6 2 0 2 の情報を参照することで非稼働中の端末装置を認識可能であるが、Network Map 表示部分 6 2 0 1 の該当する機器の表示を低い輝度あるいは明度で描画する等の G U I による表現も可能である。ユーザは個々の機器をポインティングデバイスによって Drag & Drop することで、独自に接続先の編集を行うことも可能である。

【 0 0 9 3 】

以下フローチャートを用いて、本実施形態の動作を説明する。

図 6 3 は本実施形態の稼働状況検出モジュール 1 1 1 1 が I C M P エコーリクエストを送受信する処理を示すフローチャートである。

稼働状況検出モジュール 1 1 1 1 は、オートディスカバリモジュール 1 1 1 3 からの稼働状況チェック要求を待ち(ステップ 6 3 0 1)、稼働状況チェック要求として I P アドレスを受信すると(ステップ 6 3 0 2)、I P アドレスで指定された機器に Ping (I C M P エコーリクエストメッセージ)を送信する(ステップ 6 3 0 3)。

Pingのタイムアウト内にICMPエコーリプライメッセージを受信するかどうかをチェックし(ステップ6304)、エコーリプライメッセージを受信した場合にはオートディスカバリモジュール1113にTrueを返し(ステップ6305)、それ以外の場合はFalseを返す(ステップ6306)。

ステップ6305あるいはステップ6306終了後、ステップ6301から処理を繰り返す。

稼動状況検出モジュール1111はPingの応答の有無により機器の稼動状況を検出する。

【0094】

図64は本実施形態のMIBアクセスモジュール1112がPDU(Protocol Data Unit)を作成し、SNMPメッセージを送受信する処理を示すフローチャートである。

MIBアクセスモジュール1112はオートディスカバリモジュール1113からのSNMP Get-Request(あるいはGet-Next/Set-Request)PDUの作成要求を待ち(ステップ6401)、SNMP Get(あるいはGet-Next/Set-Request)PDUの作成要求としてIPアドレス、コミュニティ名、オブジェクト名を受信する(ステップ6402)と、オブジェクト名をキーとして図12に示したOIDテーブル1121を検索する(ステップ6403)。

OIDテーブル1121にObject Name項目1201がオブジェクト名であるエントリが存在するかどうかをチェックし(ステップ6404)、エントリがヒットした場合、エントリのObject Identifier項目1202の値、IPアドレス、コミュニティ名をもとにSNMP Get(あるいはGet-Next/Set-Request)PDUを作成する(ステップ6405)。エントリがヒットしなかった場合は、オートディスカバリモジュール1113にエラーを返す(ステップ6410)。

【0095】

ステップ6405終了後、PDUをもとにSNMPメッセージを作成し、送信する(ステップ6406)。SNMPメッセージの応答としてSNMP (Get-Response)PDUの受信を待ち(ステップ6407)、応答を受信した場合はOIDテーブル1121のエントリのtype項目1203の値をもとにして受信した値の型変換

を行い(ステップ6408)、応答の受信に失敗した場合はオートディスカバリモジュール1113にエラーを返す(ステップ6410)。

ステップ6408の終了後、オートディスカバリモジュール1113に型変換したSNMP応答の値を返す(ステップ6409)。ステップ6409あるいはステップ6410の終了後、ステップ6401から処理を繰り返す。図64の処理は、図65～図68の処理実行時に呼び出される。

【0096】

図65(a)は、本実施形態のMIBアクセスモジュール1112が機器のMIB2サポート状況をチェックする処理を示すフローチャートである。

MIBアクセスモジュール1112は、オートディスカバリモジュール1113からのMIB2サポート状況チェック要求に対し、オブジェクト名としてsysDescrを指定し、図64に示したフローでSNMP Get-Requestメッセージの送受信処理を実行する(ステップ6501)。SNMP Get-Requestメッセージの送受信が成功かどうかチェック(エラーが返るかどうかをチェック)し(ステップ6502)、SNMP Get-Requestメッセージの送受信が成功の場合は機器がMIB2をサポートしていると判断し(ステップ6503)、SNMP Get-Requestメッセージの送受信が失敗の場合は機器がMIB2を非サポートであると判断し(ステップ6504)、オートディスカバリモジュール1113にMIB2サポート状況の情報を返す。

ステップ6503あるいはステップ6504の終了後、ステップ6501から処理を繰り返す。

【0097】

図65(b)は本実施形態のMIBアクセスモジュール1112が機器のIPフォワーディング機能の有無をチェックする処理を示すフローチャートである。

MIBアクセスモジュール1112は、オートディスカバリモジュール1113からのIPフォワーディング機能チェック要求に対し、オブジェクト名としてipForwardingを指定し、図64に示したフローでSNMP Get-Requestメッセージの送受信処理を実行する(ステップ6511)。SNMP Get-Requestメッセージの送受信が成功かどうかチェック(エラーが返るかどうかをチェック)し(ステップ6512)、SNMP Get-Requestメッセージの送受信が成功の場合はipForwardingの値

が“1”(True)かどうかをチェックし(ステップ6513)、SNMP Get-Requestメッセージの送受信が失敗の場合は機器がIPフォワーディング機能を保持していないと判断する(ステップ6515)。

ステップ6513の終了後、ipForwardingの値が“1”(True)の場合は、機器がIPフォワーディング機能を保持していると判断し(ステップ6514)、ipForwardingの値が“0”(False)の場合は機器がIPフォワーディング機能を保持していないと判断する(ステップ6515)。ステップ6514あるいはステップ6515の終了後、オートディスカバリモジュール1113に機器のIPフォワーディング機能の有無の情報を返し、ステップ6511から処理を繰り返す。

【0098】

図66は、本実施形態のMIBアクセスモジュール1112が機器のブリッジMIBサポート状況をチェックする処理を示すフローチャートである。

MIBアクセスモジュール1112は、オートディスカバリモジュール1113からのブリッジMIBサポート状況チェック要求に対し、オブジェクト名としてdot1dBaseBridgeAddress(ブリッジMIBの任意の実装オブジェクトで可)を指定し、図64に示したフローでSNMP Get-Requestメッセージの送受信処理を実行する(ステップ6601)。SNMP Get-Requestメッセージの送受信が成功かどうかチェック(エラーが返るかどうかをチェック)し(ステップ6602)、SNMP Get-Requestメッセージの送受信が成功の場合は機器がブリッジMIBをサポートしていると判断し(ステップ6603)、SNMP Get-Requestメッセージの送受信が失敗の場合は機器がブリッジMIBを非サポートであると判断し(ステップ6604)、オートディスカバリモジュール1113にブリッジMIBサポート状況の情報を返す。ステップ6603あるいはステップ6604の終了後、ステップ6601から処理を繰り返す。

【0099】

図67は本実施形態のMIBアクセスモジュール1112が機器のリピータMIBサポート状況をチェックする処理を示すフローチャートである。

MIBアクセスモジュール1112は、オートディスカバリモジュール1113からのリピータMIBサポート状況チェック要求に対し、オブジェクト名とし

てrpitrGroupCapacity(リピータMIBの任意の実装オブジェクトで可)を指定し、図64に示したフローでSNMP Get-Requestメッセージの送受信処理を実行する(ステップ6701)。SNMP Get-Requestメッセージの送受信が成功かどうかチェック(エラーが返るかどうかをチェック)し(ステップ6702)、SNMP Get-Requestメッセージの送受信が成功の場合は機器がリピータMIBをサポートしていると判断し(ステップ6703)、SNMP Get-Requestメッセージの送受信が失敗の場合は機器がリピータMIBを非サポートであると判断し(ステップ6704)、オートディスカバリモジュール1113にリピータMIBサポート状況の情報を返す。ステップ6703あるいはステップ6704の終了後、ステップ6701から処理を繰り返す。

【0100】

図68は本実施形態のMIBアクセスモジュール1112が機器のプリンタMIBサポート状況をチェックする処理を示すフローチャートである。

MIBアクセスモジュール1112は、オートディスカバリモジュール1113からのプリンタMIBサポート状況チェック要求に対し、オブジェクト名としてprtGeneralConfigChanges(プリンタMIBの任意の実装オブジェクトで可)を指定し、図64に示したフローでSNMP Get-Requestメッセージの送受信処理を実行する(ステップ6801)。SNMP Get-Requestメッセージの送受信が成功かどうかチェック(エラーが返るかどうかをチェック)し(ステップ6802)、SNMP Get-Requestメッセージの送受信が成功の場合は機器がプリンタMIBをサポートしていると判断し(ステップ6803)、SNMP Get-Requestメッセージの送受信が失敗の場合は機器がプリンタMIBを非サポートであると判断し(ステップ6804)、オートディスカバリモジュール1113にプリンタMIBサポート状況の情報を返す。ステップ6803あるいはステップ6804の終了後、ステップ6801から処理を繰り返す。

【0101】

図69は本実施形態のオートディスカバリモジュール1113がATテーブル1122を作成する処理を示すフローチャートである。

オートディスカバリモジュール1113はATテーブル作成要求を待ち(ステ

ップ6901)、ATテーブル作成要求として探索するネットワークのIPアドレスの範囲を指定されると(ステップ6902)、指定されたネットワークの範囲に含まれるすべてのIPアドレスの探索を開始する。未探索のIPアドレスがあるかどうかチェックし(ステップ6903)、未探索のIPアドレスがない場合はステップ6901から処理を繰り返し、未探索のIPアドレスがある場合はIPアドレスを指定して図65(a)に示したMIBアクセスモジュール1112のMIB2サポート状況チェック処理を実行する(ステップ6904)。MIBアクセスモジュール1112の返り値からIPアドレスで指定された機器がMIB2をサポートしているかどうかチェックし(ステップ6905)、MIB2をサポートしている場合はipNetMediaPhysAddressをキーに図64に示したSNMP Get-Nextメッセージ送受信処理を実行し(ステップ6906)、ipNetToMediaNetAddressをキーに図64に示したSNMP Get-Nextメッセージ送受信処理を実行する(ステップ6907)。

【0102】

機器がMIB2をサポートしていない場合はステップ6903から処理を繰り返す。ステップ6907の終了後、ステップ6906とステップ6907のSNMP Get-Nextメッセージ送受信処理が2回とも成功したかどうかチェックし(ステップ6908)、2回とも成功した場合はATテーブル1122のIP Address項目にipNetToMediaNetAddressの値を、Mac Address項目にipNetMediaPhysAddressの値を設定したエントリをATテーブル1122に追加する(ステップ6909)。SNMP Get-Nextメッセージ送受信処理の失敗があった場合、あるいはステップ6909の終了後、ステップ6903から処理を繰り返す。

【0103】

図70は、本実施形態のオートディスカバリモジュール1113がTIテーブル1123を作成する処理を示すフローチャートである。

オートディスカバリモジュール1113はTIテーブル作成要求を待ち(ステップ7001)、TIテーブル作成要求として探索するネットワークのIPアドレスの範囲を指定されると(ステップ7002)、指定されたネットワークの範囲に含まれるすべてのIPアドレスの探索を開始する。未探索のIPアドレスがあ

るかどうかチェックし(ステップ7003)、未探索のIPアドレスがない場合はステップ7001から処理を繰り返し、未探索のIPアドレスがある場合はIPアドレスを指定して図71に示すTIテーブルの各項目の値を取得する処理を実行する(ステップ7004)。ステップ7004の終了後、TIテーブル1123に新規エントリを追加し(ステップ7005)、ステップ7003から処理を繰り返す。

【0104】

図71は、本実施形態のオートディスカバリモジュール1113がTIテーブル1123の作成時にTIテーブル1123の各項目の値を取得する処理を示すフローチャートである。

オートディスカバリモジュール1113はTIテーブル1123の各項目の値を取得要求を待ち(ステップ7101)、TIテーブル1123の各項目の値を取得要求として探索するIPアドレスを受信すると(ステップ7102)、IPアドレスをキーにATテーブル1122のIP Addressを検索し、取得したエントリのMac Address項目の値をTIテーブル1123のMac Address項目に設定する(ステップ7103)。

次に、IPアドレスをキーに機器のホスト名を解決し、TIテーブル1123のHost Name項目にホスト名を設定する(ステップ7104)。次に、IPアドレスをキーに図63に示した稼動状況チェック処理を実行し(ステップ7105)、TIテーブル1123のalive項目に稼動状況チェック処理の返り値を設定する。次に、図65(a)に示したMIB2サポート状況チェック処理を実行し(ステップ7106)、TIテーブル1123のMIB2項目にMIB2サポート状況チェック処理の返り値を設定する。

【0105】

次に、図65(b)に示したIPフォワーディング機能チェック処理を実行し(ステップ7107)、TIテーブル1123のforwarding項目にIPフォワーディング機能チェック処理の返り値を設定する。次に、図66に示したブリッジMIBサポート状況チェック処理を実行し(ステップ7108)、TIテーブル1123のbridge項目にブリッジMIBサポート状況チェック処理の返り値を設定す

る。

次に、図 6 7 に示したリピータ M I B サポート状況チェック処理を実行し(ステップ 7 1 0 9)、T I テーブル 1 1 2 3 の repeater 項目にリピータ M I B サポート状況チェック処理の戻り値を設定する。次に、図 6 8 に示したプリンタ M I B サポート状況チェック処理を実行し(ステップ 7 1 1 0)、T I テーブル 1 1 2 3 の printer 項目にプリンタ M I B サポート状況チェック処理の戻り値を設定する。次に、図 7 2 に示す機器タイプ認識処理を実行し(ステップ 7 1 1 1)、T I テーブル 1 1 2 3 の type 項目に機器タイプ認識処理の戻り値を設定する。ステップ 7 1 1 1 の終了後、ステップ 7 1 0 1 から処理を繰り返す。

【 0 1 0 6 】

図 7 2 は、本実施形態のオートディスカバリモジュール 1 1 1 3 が T I テーブル 1 1 2 3 の作成時に機器タイプを認識する処理を示すフローチャートである。

オートディスカバリモジュール 1 1 1 3 は、機器タイプの認識要求を待ち(ステップ 7 2 0 1)、機器タイプの認識要求として T I テーブル 1 1 2 3 の該当するエントリの forwarding 項目、bridge 項目、repeater 項目、printer 項目の値を受信する(ステップ 7 2 0 2)と、forwarding 項目の値が “ 1 ” (True) かどうかチェックする(ステップ 7 2 0 3)。

forwarding 項目の値が “ 1 ” (True) の場合は、bridge 項目の値が “ 1 ” (True) かどうかチェックする(ステップ 7 2 0 4)。

forwarding 項目の値が “ 1 ” (True) で、bridge 項目の値が “ 1 ” (True) の場合は機器をルータと認識する(ステップ 7 2 0 5)。forwarding 項目の値が “ 1 ” (True) で、bridge 項目の値が “ 0 ” (False) の場合は機器をスイッチングハブと認識する(ステップ 7 2 0 6)。

ステップ 7 2 0 3 で forwarding 項目の値が “ 0 ” (False) の場合は、bridge 項目の値が “ 1 ” (True) かどうかチェックする(ステップ 7 2 0 7)。bridge 項目の値が “ 1 ” (True) の場合は、repeater 項目の値が “ 1 ” (True) かどうかチェックする(ステップ 7 2 0 8)。forwarding 項目の値が “ 0 ” (False) で、bridge 項目の値が “ 1 ” (True) で repeater 項目の値が “ 1 ” (True) の場合は機器をスイッチングハブと認識する(ステップ 7 2 0 6)。

【 0 1 0 7 】

forwarding項目の値が“0”(True)でbridge項目の値が“1”(True)でrepeater項目の値が“0”(False)の場合は機器をブリッジと認識する(ステップ7209)。ステップ7207でbridge項目の値が“0”(False)の場合はrepeater項目の値が“1”(True)かどうかチェックする(ステップ7210)。forwarding項目の値が“0”(False)でbridge項目の値が“0”(False)でrepeater項目の値が“1”(True)の場合は機器をインテリジェントハブと認識する(ステップ7211)。

ステップ7210でrepeater項目の値が“0”(False)の場合はprinter項目の値が“1”(True)かどうかチェックする(ステップ7212)。forwarding項目の値が“0”(False)でbridge項目の値が“0”(False)でrepeater項目の値が“0”(False)でprinter項目の値が“1”(True)の場合は機器をプリンタと認識する(ステップ7213)。forwarding項目の値が“0”(False)でbridge項目の値が“0”(False)でrepeater項目の値が“0”(False)でprinter項目の値が“0”(False)の場合は機器を端末装置と認識する(ステップ7214)。ステップ7205、ステップ7206、ステップ7209、ステップ7211、ステップ7213、ステップ7214の任意のステップ終了後、ステップ7201から処理を繰り返す。

【 0 1 0 8 】

図73は本実施形態のオートディスカバリモジュール1113がPFテーブル1124を作成する処理を示すフローチャートである。

オートディスカバリモジュール1113はPFテーブル1124の作成要求を待ち(ステップ7301)、PFテーブル1124の作成要求を受信する(ステップ7302)と、TIテーブル1123のすべてのエントリの検索を開始する。TIテーブル1123に未探索のエントリがあるかどうかチェックし(ステップ7303)、TIテーブル1123に未探索のエントリがない場合はステップ7301から処理を繰り返し、TIテーブル1123に未探索のエントリがある場合はTIテーブル1123の該当するエントリのbridge項目の値が“1”(True)かどうかチェックする(ステップ7304)。bridge項目の値が“1”(True)の場

合は図 7 4 に示すブリッジ M I B サポート機器に対する処理を実行する(ステップ 7 3 0 5)。bridge 項目の値が “ 0 ” (False) の場合は、T I テーブル 1 1 2 3 の該当するエントリの repeater 項目の値が “ 1 ” (True) かどうかチェックする(ステップ 7 3 0 6)。

【 0 1 0 9 】

repeater 項目の値が “ 1 ” (True) の場合は図 7 5 に示すリピータ M I B サポート機器に対する処理を実行する(ステップ 7 3 0 7)。bridge 項目の値が “ 0 ” (False) の場合は T I テーブル 1 1 2 3 の該当するエントリの M I B 2 項目の値が “ 1 ” (True) かどうかチェックする(ステップ 7 3 0 8)。M I B 2 項目の値が “ 1 ” (True) の場合は図 7 6 に示すインターフェース M I B サポート機器に対する処理を実行する(ステップ 7 3 0 9)。M I B 2 項目の値が “ 0 ” (False) の場合はステップ 7 3 0 3 から処理を繰り返す。ステップ 7 3 0 5、ステップ 7 3 0 7、ステップ 7 3 0 9 の任意のステップ終了後、ステップ 7 3 0 3 から処理を繰り返す。

【 0 1 1 0 】

図 7 4 は本実施形態のオートディスカバリモジュール 1 1 1 3 が P F テーブル 1 1 2 4 の作成時にブリッジ M I B サポート機器に対して実行する処理を示すフローチャートである。

オートディスカバリモジュール 1 1 1 3 はブリッジ M I B サポート機器に対する処理要求を待ち(ステップ 7 4 0 1)、ブリッジ M I B サポート機器に対する処理要求として T I テーブル 1 1 2 3 の IP Address 項目の値を受信し、P F テーブル 1 1 2 4 の Source IP Address 項目に設定する(ステップ 7 4 0 2)と、IP Address 項目の値をキーに A T テーブル 1 1 2 2 の IP Address 項目を検索し、ヒットしたエントリの Mac Address 項目の値を P F テーブル 1 1 2 4 の Source Mac Address 項目に設定する(ステップ 7 4 0 3)。次に、I P アドレスで指定された機器に対して未探索フォワーディング情報があるかどうかチェック (SNMP Get-Next メッセージ送受信がエラーになるまで処理を実行)し(ステップ 7 4 0 4)、未探索フォワーディング情報がない場合はステップ 7 4 0 1 から処理を繰り返す。未探索フォワーディング情報がある場合はオブジェクト名として dot1dTpFdbAddress を指定し、図 6 4 に示したフローで SNMP Get-Next メッセージの送受信処理を実

行し、返り値を P F テーブル 1 1 2 4 の Destination Mac Address 項目に設定する (ステップ 7 4 0 5)。

【 0 1 1 1 】

同様に、オブジェクト名として dot1dTpFdbPort を指定し、図 6 4 に示すフローで SNMP Get-Next メッセージの送受信処理を実行し、返り値を P F テーブル 1 1 2 4 の Source Port 項目に設定する (ステップ 7 4 0 6)。次に、設定した Destination Mac Address 項目の値をキーに A T テーブル 1 1 2 2 の Mac Address 項目を検索し、ヒットしたエントリの IP Address 項目の値を P F テーブル 1 1 2 4 の Destination IP Address 項目に設定する (ステップ 7 4 0 7)。最後に、P F テーブル 1 1 2 4 に新規エントリを追加し (ステップ 7 4 0 8)、ステップ 7 4 0 4 から処理を繰り返す。

【 0 1 1 2 】

図 7 5 は、本実施形態のオートディスカバリモジュール 1 1 1 3 が P F テーブル 1 1 2 4 の作成時にリピータ M I B サポート機器に対して実行する処理を示すフローチャートである。

オートディスカバリモジュール 1 1 1 3 は、リピータ M I B サポート機器に対する処理要求を待ち (ステップ 7 5 0 1)、リピータ M I B サポート機器に対する処理要求として T I テーブル 1 1 2 3 の IP Address 項目の値を受信し、P F テーブル 1 1 2 4 の Source IP Address 項目に設定する (ステップ 7 5 0 2) と、IP Address 項目の値をキーに A T テーブル 1 1 2 2 の IP Address 項目を検索し、ヒットしたエントリの Mac Address 項目の値を P F テーブル 1 1 2 4 の Source Mac Address 項目に設定する (ステップ 7 5 0 3)。

次に、SNMP Get-Next メッセージ送受信のアクセス回数にあらかじめ閾値を設定しておき、アクセス回数が閾値を超えているかどうかチェックし (ステップ 7 5 0 4)、アクセス回数が閾値を超えている場合は図 7 7 に示すフォワーディング情報予測処理を実行する (ステップ 7 5 0 9)。アクセス回数が閾値を超えない場合は rpPtrAddrTrackLastSourceAddrChanges を指定して図 6 4 に示した SNMP Get-Next メッセージ送受信処理を実行する (ステップ 7 5 0 5)。

【 0 1 1 3 】

ステップ7505の終了後、SNMP Get-Nextメッセージ送受信処理の返り値であるrptrAddrTrackLastSourceAddrChangesの値を記憶しておき、前回アクセス時と値の比較を行い、オブジェクトの値に変更があるかどうかチェックする(ステップ7506)。オブジェクトの値に変更がない場合は処理を一時停止(Sleep処理)し(ステップ7507)、アクセス回数が閾値を超えるまでステップ7504から処理を繰り返す。ステップ7506でオブジェクトの値に変更がある場合は、現在実行中のスレッドとは別のスレッドを作成し、作成したスレッド内で図76に示すフォワーディング情報学習処理を開始する(ステップ7508)。

ステップ7508、ステップ7509の任意のステップ終了後、ステップ7501から処理を繰り返す。フォワーディング情報学習処理は、リピータMIBの仕様がRFCに準拠した機器に対し、一定間隔でリピータMIBにアクセスして情報を収集する処理であり、フォワーディング情報予測処理はリピータMIBの仕様がRFCに準拠していない機器に対し、インタフェースMIBを利用してフォワーディング情報を検出する処理である。

【0114】

図76は、オートディスカバリモジュール1113がPFテーブル1124の作成時にフォワーディング情報を学習する処理を示すフローチャートである。

オートディスカバリモジュール1113は、フォワーディング情報学習処理要求を待ち(ステップ7601)、フォワーディング情報学習処理要求としてTIテーブル1123のIP Address項目の値を受信し、PFテーブル1124のSource IP Address項目に設定する(ステップ7602)と、IPアドレスで指定された機器の全ポートの探索が終了しているかどうかチェックする(ステップ7603)。全ポートの探索が終了している場合は、ステップ7601から処理を繰り返し、全ポートの探索が終了していない場合はrptrAddrTrackLastSourceAddressをキーに指定して図64に示したSNMP Get-Nextメッセージ送受信処理を実行し、SNMP Get-Nextメッセージ送受信処理の返り値をPFテーブル1124のDestination Mac Address項目に設定する(ステップ7604)。

【0115】

次に、設定したDestination Mac Address項目の値が既に検出済みであるかチ

エックし(ステップ7605)、検出済みである場合はステップ7603から処理を繰り返し、検出済みでない場合はrptraAddrTrackPortIndexをキーに指定して図64に示したSNMP Get-Nextメッセージ送受信処理を実行し、戻り値をPFテーブル1124のSource Port項目に設定する(ステップ7606)。

【0116】

次に、設定したDestination Mac Address項目の値をキーにATテーブル1122のMac Address項目を検索し、ヒットしたエントリのIP Address項目の値をPFテーブル1124のDestination IP Address項目に設定する(ステップ7607)。最後に、PFテーブル1124に新規エントリを追加し(ステップ7608)、ステップ7603から処理を繰り返す。

【0117】

図77は、オートディスカバリモジュール1113がPFテーブル1124の作成時にフォワーディング情報を予測する処理を示すフローチャートである。

オートディスカバリモジュール1113は、フォワーディング情報予測処理要求を待ち(ステップ7701)、フォワーディング情報予測処理要求としてTIテーブル1123のIP Address項目の値を受信し、PFテーブル1124のSource IP Address項目に設定する(ステップ7702)と、IPアドレスで指定された機器の全ポートの探索が終了しているかどうかチェックする(ステップ7703)。全ポートの探索が終了している場合は、ステップ7701から処理を繰り返し、全ポートの探索が終了していない場合はrptraAddrTrackLastSourceAddressをキーに指定して図64に示したNMP Get-Nextメッセージ送受信処理を実行し、SNMP Get-Nextメッセージ送受信処理の戻り値をPFテーブル1124のDestination Mac Address項目に設定する(ステップ7704)。

rptraAddrTrackPortIndexをキーに指定して図64に示したSNMP Get-Nextメッセージ送受信処理を実行し、戻り値をPFテーブル1124のSource Port項目に設定する(ステップ7705)。

【0118】

次に、設定したDestination Mac Address項目の値をキーにATテーブルのMac Address項目を検索し、ヒットしたエントリのIP Address項目の値をPFテーブ

ル 1 1 2 4 の Destination IP Address 項目に設定する(ステップ 7 7 0 6)。次に、P F テーブル 1 1 2 4 に新規エントリを追加する(ステップ 7 7 0 7)。P F テーブル 1 1 2 4 に 1 件のエントリを追加後、rptraAddrTrackLastSourceAddrChanges をキーに指定して図 6 4 に示した SNMP Get-Next メッセージ送受信処理を実行し(ステップ 7 7 0 8)、SNMP Get-Next メッセージ送受信処理の戻り値である rptraAddrTrackLastSourceAddrChanges の値が “1” より大きいチェックする(ステップ 7 7 0 9)。

rptraAddrTrackLastSourceAddrChanges の値が “1” より大きい場合は P F テーブル 1 1 2 4 に残りのエントリを追加するために M I B 2(interfaces M I B) サポート機器に対して実行する処理を実行し(ステップ 7 7 1 0)、rptraAddrTrackLastSourceAddrChanges の値が “1” 以下の場合はステップ 7 7 0 3 から処理を繰り返す。ステップ 7 7 1 0 の終了後も同様にステップ 7 7 0 3 から処理を繰り返す。

【 0 1 1 9 】

図 7 8 は、オートディスカバリモジュール 1 1 1 3 が P F テーブル 1 1 2 4 の作成時に M I B 2(interfaces M I B) サポート機器に対して実行する処理を示すフローチャートである。

オートディスカバリモジュール 1 1 1 3、は M I B 2(interfaces M I B) サポート機器に対して実行する処理要求を待ち(ステップ 7 8 0 1)、M I B 2 サポート機器に対して実行する処理要求として T I テーブル 1 1 2 3 の IP Address 項目の値を受信し、P F テーブル 1 1 2 4 の Source IP Address 項目に設定する(ステップ 7 8 0 2)と、図 7 9 に示す管理者端末の接続ポート検出処理を実行する(ステップ 7 8 0 3)。

次に、図 8 0 に示す管理者端末以外の機器の接続ポート検出処理を実行する(ステップ 7 8 0 4)。最後に、P F テーブル 1 1 2 4 に新規エントリを追加し(ステップ 7 8 0 5)、ステップ 7 8 0 1 から処理を繰り返す。

【 0 1 2 0 】

図 7 9 は、オートディスカバリモジュール 1 1 1 3 が P F テーブル 1 1 2 4 の作成時に管理者端末 7 1 の接続ポートを検出する処理を示すフローチャートであ

る。

オートディスカバリモジュール1113は管理者端末71の接続ポートを検出処理要求を待ち(ステップ7901)、管理者端末71の接続ポートを検出処理要求としてネットワーク中継装置のIPアドレス値を受信する(ステップ7902)と、IPアドレスで指定されたネットワーク中継装置の全ポートの探索が終了しているかどうかチェックする(ステップ7903)。全ポートの探索が終了している場合は配列alive[ポート番号]の値が“0”(False)であるポート番号を返す(ステップ7906)。全ポートの探索が終了していない場合はifAdminStatusをキーに、値を“0”(False)と指定して図64に示したSNMP Set-Requestメッセージ送受信処理を実行し、SNMP管理プロトコルを利用して該当するポートを塞ぐ(ステップ7904)。

IPアドレスで指定されたネットワーク中継装置を指定して図63に示したICMPエコーリクエストの送受信処理を実行し、戻り値が“1”(True)の場合は、alive[ポート番号]変数に“1”を設定し、戻り値が“0”(False)の場合はalive[ポート番号]変数にを設定する(ステップ7905)。ただし、alive[ポート番号]の初期値は“0”(False)とする。

ステップ7905の終了後、ステップ7903から処理を繰り返す。ステップ7906の終了後、ステップ7901から処理を繰り返す。管理者端末71が接続しているポート以外のポートを塞いだ場合は、管理者端末71からネットワーク中継装置へのICMPエコーリクエスト送受信処理が成功するが、管理者端末71が接続しているポート塞いだ場合は管理者端末71からネットワーク中継装置へのICMPエコーリクエスト送受信処理の応答が返らないことを利用している。

【0121】

図80は、オートディスカバリモジュール1113がPFテーブル1124の作成時に管理者端末以外の機器の接続ポートを検出する処理を示すフローチャートである。

オートディスカバリモジュール1113は、管理者端末71以外の機器の接続ポート検出処理要求を待ち(ステップ8001)、管理者端末71以外の機器の接

続ポート検出処理要求としてネットワーク中継装置のIPアドレス値と管理者端末71が接続しているネットワーク中継装置上のポート番号を受信する(ステップ8002)と、TIテーブル1123の探索を開始し、未探索の機器があるかチェックする(ステップ8003)。未探索の機器がある場合には、TIテーブル1123のエントリのalive項目の値をpre_alive変数に設定し(ステップ8004)、未探索の機器がない場合にはステップ8001から処理を繰り返す。

【0122】

ステップ8004の終了後、IPアドレスで指定されたネットワーク中継装置の全ポートの探索が終了しているかどうかチェックする(ステップ8005)。全ポートの探索が終了している場合はpre_alive変数が“1”(True)でかつalive[ポート番号]が“0”(False)となるポート番号が存在するかチェックする(ステップ8008)。全ポートの探索が終了していない場合はifAdminStatusをキーに、値を“0”(False)と指定して図64に示したSNMP Set-Requestメッセージ送受信処理を実行し、SNMP管理プロトコルを利用して該当するポートを塞ぎ(ステップ8006)、IPアドレスで指定されたネットワーク中継装置を指定して図63に示したICMPエコーリクエストの送受信処理を実行し、戻り値が“1”(True)の場合はalive[ポート番号]変数に“1”を設定し、戻り値が“0”(False)の場合はalive[ポート番号]変数に“0”を設定する(ステップ8007)。ただしalive[ポート番号]の初期値は“0”(False)とする。

【0123】

ステップ8007の終了後、ステップ8005から処理を繰り返す。ステップ8008の終了後、条件を満たすポートが見つからない場合は、管理者端末71の接続ポート番号を返し(ステップ8009)、条件を満たすポートが見つかった場合はalive[ポート番号]変数が“0”(False)となるポート番号を返す(ステップ8010)。ステップ8009、ステップ8010の任意のステップ終了後、ステップ8001から処理を繰り返す。

ネットワーク中継装置の特定のポートを塞ぐことで、任意の機器へのICMPエコーリクエストの応答が受信できなくなる場合、該当するポートが接続ポートとなる。

【0124】

図81は、オートディスカバリモジュール1113がTSテーブル1125を作成する処理を示すフローチャートである。

オートディスカバリモジュール1113はPFテーブル1124の作成要求を待ち(ステップ8101)、PFテーブル1124の作成要求を受信すると(ステップ8102)、図82に示すRoot装置決定処理を実行し、Root装置のIPアドレスをRoot変数に設定し、Unitsリスト変数の全項目を削除して初期化する(ステップ8103)。

次に、図83に示すネットワーク中継装置間の接続の決定処理を実行する(ステップ8104)。次に、図114に示すネットワーク中継装置と端末装置の接続の決定処理を実行する(ステップ8105)。最後に、図115に示すインタフェースMIB評価処理を実行し(ステップ8106)、ステップ8101から処理を繰り返す。

【0125】

図82は、オートディスカバリモジュール1113がTSテーブル1125の作成時にRoot装置を決定する処理を示すフローチャートである。

オートディスカバリモジュール1113はRoot装置決定処理要求を待ち(ステップ8201)、Root装置決定処理要求を受信すると(ステップ8202)、TIテーブル1123の探索を開始し、未探索の機器があるかチェックする(ステップ8203)。未探索の機器がない場合には、ステップ8201から処理を繰り返す。未探索の機器がある場合には、TIテーブル1123の該当するエントリのtype項目の値がR(Routerを示す識別子)かチェックし(ステップ8204)、type項目の値がR以外の場合にはステップ8203から処理を繰り返す。type項目の値がRの場合にはRoot変数にルータのIPアドレスを追加する(ステップ8205)。最後に、図103示すTSテーブル1125に対するRootエントリ追加処理を実行する(ステップ8206)。ステップ8206の終了後、ステップ8201から処理を繰り返す。

【0126】

図83は、オートディスカバリモジュール1113がTSテーブル1125の

作成時にネットワーク中継装置間の接続を決定する処理を示すフローチャートである。

オートディスカバリモジュール 1 1 1 3 はネットワーク中継装置間の接続を決定処理要求を待ち(ステップ 8 3 0 1)、ネットワーク中継装置間の接続の決定処理要求を受信すると(ステップ 8 3 0 2)、Units リスト変数に P F テーブル 1 1 2 4 の全エントリの Source IP Address 項目の値の中で、Root 変数と同一のものを除くすべての値を追加する(ステップ 8 3 0 3)。

次に、Units リスト変数の要素の中から任意の 2 つの要素の組合せの集合を選択し、未探索の組合せ(Unit1 変数、Unit2 変数に設定)があるかチェックする(ステップ 8 3 0 4)。未探索の組合せがある場合は図 8 4 に示す接続モデルの決定処理を実行し(ステップ 8 3 0 5)、図 1 0 2 に示す T S テーブル 1 1 2 5 に対するエントリの追加処理を実行(ステップ 8 3 0 6)した後でステップ 8 3 0 4 から処理を繰り返す。接続モデルの決定処理では図 2 2 ～図 4 5 に示した形式で Unit 1 と Unit 2 に関する接続モデルを決定する。T S テーブル 1 1 2 5 に対するエントリの追加処理では、決定した接続モデルごと決まったフォーマットで T S テーブル 1 1 2 5 にエントリを格納している。

ステップ 8 3 0 4 で未探索の組合せがない場合は図 1 0 7 に示す親子関係の決定処理を実行し(ステップ 8 3 0 7)、ステップ 8 3 0 1 から処理を繰り返す。親子関係の決定処理では、T S テーブル 1 1 2 5 に親子関係がある機器同士のエントリだけを抽出し、T S テーブル 1 1 2 3 の最終形を決定する。

【 0 1 2 7 】

図 8 4 はオートディスカバリモジュール 1 1 1 3 による T S テーブル 1 1 2 5 の作成時の接続モデルの決定処理を示すフローチャートである。

オートディスカバリモジュール 1 1 1 3 は接続モデルの決定処理要求を待ち(ステップ 8 4 0 1)、接続モデルの決定処理要求を受信すると(ステップ 8 4 0 2)、図 8 5 に示す方法で I P アドレスが Unit1 変数に等しい機器に関するネットワーク機器の分類処理を実行する(ステップ 8 4 0 3)。同様に、図 8 5 に示す方法で I P アドレスが Unit2 変数に等しい機器に関するネットワーク機器の分類処理を実行する(ステップ 8 4 0 4)。ネットワーク機器の分類処理では図 2 1 に

示した分類を決定する。最後に接続検出条件チェック処理を実行(ステップ8405)後、ステップ8401から処理を繰り返す。接続検出条件チェック処理では図46に示した接続検出条件のチェックを実行する。

【0128】

図85は、オートディスカバリモジュール1113によるTSテーブル1125の作成時のネットワーク機器の分類処理を示すフローチャートである。

オートディスカバリモジュール1113はネットワーク機器の分類処理要求を待ち(ステップ8501)、ネットワーク機器の分類処理要求を受信すると(ステップ8502)、PFテーブル1124のすべてのエントリの中からSource IP Address項目の値がUnit1もしくはUnit2に等しく、Destination IP Address項目の値がRoot変数値に等しいエントリを検索する(ステップ8503)。

検索したエントリが存在するかチェックする(ステップ8504)。ステップ8504で検索したエントリが存在する場合、Unitsリスト変数に含まれるすべてネットワーク中継装置のエントリを順番にTarget変数に設定し、Target変数が未検索のものであるかどうかチェックする(ステップ8505)。

【0129】

ステップ8504で検索したエントリが存在しない場合は、ステップ8503でSource IP Address項目の値がUnit1に等しい場合はCategory1変数にSFを設定し、ステップ8503でSource IP Address項目の値がUnit2に等しい場合はCategory2変数にSFを設定する。

ステップ8505でTarget変数が未検索の場合はPFテーブル1124のすべてのエントリの中からSource IP Address項目の値がUnit1もしくはUnit2でDestination IP Address項目の値がTarget変数に等しいエントリを検索する(ステップ8506)。

ステップ8505で未検索のTarget変数が存在しない場合はステップ8501から繰り返す。次に、ステップ8506の検索項目があるかチェックする(ステップ8507)。ステップ8507の検索項目がある場合は、ステップ8503でSource IP Address項目の値がUnit1に等しい場合はCategory1変数にCFを設定し、ステップ8503でSource IP Address項目の値がUnit2に等しい場合はCate

gory2変数にCFを設定する。

ステップ 8 5 0 7 の検索項目がない場合は、ステップ 8 5 0 3 で Source IP Address項目の値がUnit1に等しい場合はCategory1変数にCFを設定し、ステップ 8 5 0 3 で Source IP Address項目の値がUnit2に等しい場合はCategory2変数にCFを設定する。

Unitsリスト変数に含まれるすべての機器への接続情報がある場合はCFが設定され、1台でも接続情報が含まれていない場合はIFが設定される。

【 0 1 3 0 】

図 8 6 はオートディスカバリモジュール 1 1 1 3 による T S テーブル 1 1 2 5 の作成時の接続検出条件のチェック処理を示すフローチャートである。

オートディスカバリモジュール 1 1 1 3 は接続検出条件のチェック処理要求を待ち(ステップ 8 6 0 1)、接続検出条件のチェック処理要求として、2台のネットワーク機器のIPアドレスを格納したUnit1変数、Unit2変数と2台のネットワーク機器の分類を格納したCategory1変数、Category2変数を受信すると(ステップ 8 6 0 2)、Category1変数がCFに等しく、Category2変数がCFに等しいかチェックする(ステップ 8 6 0 3)。

ステップ 8 6 0 3 でCategory1変数がCFに等しく、Category2変数がCFに等しい場合は、図 8 7 に示す集合 (R, CF, CF) の接続検出条件チェック処理を実行し(ステップ 8 6 0 4)、ステップ 8 6 0 1 から繰り返す。

ステップ 8 6 0 2 でCategory1変数がCFに等しく、Category2変数がCFに等しいという条件を満たしていない場合は、Category1変数がCFに等しく、Category2変数がIFに等しい、もしくはCategory1変数がIFに等しく、Category2変数がCFに等しいかチェックする(ステップ 8 6 0 5)。ステップ 8 6 0 5 でCategory1変数がCFに等しく、Category2変数がIFに等しい、もしくはCategory1変数がIFに等しく、Category2変数がCFに等しい場合は、図 8 8 に示す集合 (R, CF, IF) の接続検出条件チェック処理を実行し(ステップ 8 6 0 6)、ステップ 8 6 0 1 から繰り返す。

【 0 1 3 1 】

ステップ 8 6 0 5 でCategory1変数がCFに等しく、Category2変数がIFに等しい

、もしくはCategory1変数がIFに等しく、Category2変数がCFに等しいという条件を満たしていない場合は、Category1変数がCFに等しく、Category2変数がSFに等しい、もしくはCategory1変数がSFに等しく、Category2変数がCFに等しいかチェックする(ステップ8607)。ステップ8607でCategory1変数がCFに等しく、Category2変数がSFに等しい、もしくはCategory1変数がSFに等しく、Category2変数がCFに等しい場合は、図91に示す集合(R, CF, SF)の接続検出条件チェック処理を実行し(ステップ8608)、ステップ8601から繰り返す。

ステップ8607でCategory1変数がCFに等しく、Category2変数がSFに等しい、もしくはCategory1変数がSFに等しく、Category2変数がCFに等しいという条件を満たしていない場合は、Category1変数がIFに等しく、Category2変数がIFに等しいかチェックする(ステップ8609)。ステップ8609でCategory1変数がIFに等しく、Category2変数がIFに等しい場合は、図94に示す集合(R, IF, IF)の接続検出条件チェック処理を実行し(ステップ8610)、ステップ8601から繰り返す。ステップ8609でCategory1変数がIFに等しく、Category2変数がIFに等しいという条件を満たしていない場合は、Category1変数がIFに等しく、Category2変数がSFに等しい、もしくはCategory1変数がSFに等しく、Category2変数がIFに等しいかチェックする(ステップ8611)。

【0132】

ステップ8611でCategory1変数がIFに等しく、Category2変数がSFに等しい、もしくはCategory1変数がSFに等しく、Category2変数がIFに等しい場合は、図96に示す集合(R, IF, SF)の接続検出条件チェック処理を実行し(ステップ8612)、ステップ8601から繰り返す。

ステップ8611でCategory1変数がIFに等しく、Category2変数がSFに等しい、もしくはCategory1変数がSFに等しく、Category2変数がIFに等しいという条件を満たしていない場合は、Category1変数がSFに等しく、Category2変数がSFに等しいかチェックする(ステップ8613)。ステップ8613でCategory1変数がSFに等しく、Category2変数がSFに等しい場合は、図101に示す集合(R, SF, SF)の接続検出条件チェック処理を実行し(ステップ8614)、ステップ8601から繰り返す。ステップ8613でCategory1変数がSFに等しく、Category2変数

がSFに等しいという条件を満たしていない場合は、ステップ 8 6 0 1 から繰り返す。

【 0 1 3 3 】

図 8 7 は、オートディスカバリモジュール 1 1 1 3 による T S テーブル 1 1 2 5 の作成時の集合 (R, CF, CF) の接続条件チェック処理を示すフローチャートである。

オートディスカバリモジュール 1 1 1 3 は集合 (R, CF, CF) の接続条件チェック処理要求を待ち (ステップ 8 7 0 1)、集合 (R, CF, CF) の接続条件チェック処理要求を受信すると (ステップ 8 7 0 2)、P F テーブル 1 1 2 4 のすべてのエントリの中から Source IP Address 項目の値が CF1 変数 (Unit1 変数と等しい) で Destination IP Address 項目の値が Root 変数に等しいエントリを検索し、該当するエントリの Source Port 項目の値を CF1R 変数に設定する (ステップ 8 7 0 3)。

同様にして、P F テーブル 1 1 2 4 のすべてのエントリの中から Source IP Address 項目の値が CF2 変数 (Unit2 変数と等しい) で Destination IP Address 項目の値が Root 変数に等しいエントリを検索し、該当するエントリの Source Port 項目の値を CF2R 変数に設定する (ステップ 8 7 0 4)。

また、P F テーブル 1 1 2 4 のすべてのエントリの中から Source IP Address 項目の値が CF1 変数 (Unit1 変数と等しい) で Destination IP Address 項目の値が CF2 変数 (Unit2 変数と等しい) に等しいエントリを検索し、該当するエントリの Source Port 項目の値を CF1CF2 変数に設定する (ステップ 8 7 0 5)。

【 0 1 3 4 】

同様にして、P F テーブル 1 1 2 4 のすべてのエントリの中から Source IP Address 項目の値が CF2 変数 (Unit2 変数と等しい) で Destination IP Address 項目の値が CF1 変数 (Unit1 変数と等しい) に等しいエントリを検索し、該当するエントリの Source Port 項目の値を CF2CF1 変数に設定する (ステップ 8 7 0 6)。CF1R 変数の値と CF1CF2 の値を比較し (ステップ 8 7 0 7)、CF1 から見て、R と CF2 が別ポートに接続しているかチェックする (CF2R 変数と CF2CF1 変数を比較することも可能)。

ステップ 8 7 0 7 で CF1R 変数の値と CF1CF2 の値が等しい場合は、Paddr 変数に C

F2変数の値、Caddr変数にCF1変数の値、Pport変数にCF2CF1変数の値、Cport変数にCF1CF2変数の値、Model変数にR-CF-CFを設定し(ステップ8708)、ステップ8701から繰り返す。

ステップ8707でCF1R変数の値とCF1CF2の値が等しくない場合は、Paddr変数にCF1変数の値、Caddr変数にCF2変数の値、Pport変数にCF1CF2変数の値、Cport変数にCF2CF1変数の値、Model変数にR-CF-CFを設定し(ステップ8709)、ステップ8701から繰り返す。図46のR-CF-CFモデルでは接続検出条件がないことから、図87は図22に示した方法で接続関係を検出する。

【0135】

図88はオートディスカバリモジュール1113によるTSテーブル1125の作成時の集合(R, CF, IF)の接続条件チェック処理を示すフローチャートである。

オートディスカバリモジュール1113は集合(R, CF, IF)の接続条件チェック処理要求を待ち(ステップ8801)、集合(R, CF, IF)の接続条件チェック処理要求を受信すると(ステップ8802)、PFテーブル1124のすべてのエントリの中からSource IP Address項目の値がCF変数(Unit1変数とUnit2変数の内でCFと認識された機器)でDestination IP Address項目の値がRoot変数に等しいエントリを検索し、該当するエントリのSource Port項目の値をCFR変数に設定する(ステップ8803)。

同様に、PFテーブル1124のすべてのエントリの中からSource IP Address項目の値がCF変数(Unit1変数とUnit2変数の内でCFと認識された機器)でDestination IP Address項目の値がIF変数(Unit1変数とUnit2変数の内でIFと認識された機器)に等しいエントリを検索し、該当するエントリのSource Port項目の値をCFIF変数に設定する(ステップ8804)。CFR変数の値とCFIF値を比較し(ステップ8805)、CFから見て、RとIFが別ポートに接続しているかチェックする。

ステップ8805でCFR変数の値とCFIFの値が等しい場合は、Paddr変数にIF変数の値、Caddr変数にCF変数の値、Model変数にR-IF-CFを設定し、図89に示すR-IF-CFモデルの接続検出条件チェック処理を実行し(ステップ8806)、ステップ8801から繰り返す。

ステップ 8 8 0 5 で CFR 変数の値と CFIF の値が等しくない場合は、Paddr 変数に CF 変数の値、Caddr 変数に IF 変数の値、Model 変数に R-CF-IF を設定し、図 9 0 に示す R-CF-IF モデルの接続検出条件チェック処理を実行し(ステップ 8 8 0 7)、ステップ 8 8 0 1 から繰り返す。

【 0 1 3 6 】

図 8 9 はオートディスカバリモジュール 1 1 1 3 による T S テーブル 1 1 2 5 の作成時の R-IF-CF モデルの接続条件チェック処理を示すフローチャートである。

オートディスカバリモジュール 1 1 1 3 は R-IF-CF モデルの接続条件チェック処理要求を待ち(ステップ 8 9 0 1)、R-IF-CF モデルの接続条件チェック処理要求を受信すると(ステップ 8 9 0 2)、P F テーブル 1 1 2 4 のすべてのエントリの中から Source IP Address 項目の値が CF 変数(Unit1 変数と Unit2 変数の内で、図 8 8 で CF と認識された機器)で Destination IP Address 項目の値が IF 変数(Unit1 変数と Unit2 変数の内で、図 8 8 で IF と認識された機器)に等しいエントリを検索し、該当するエントリの Source Port 項目の値を CFIF 変数に設定する(ステップ 8 9 0 3)。

【 0 1 3 7 】

同様に、P F テーブル 1 1 2 4 におけるすべてのエントリの中から Source IP Address 項目の値が CF 変数(Unit1 変数と Unit2 変数の内で CF と認識された機器)で Source Port 項目の値が CFIF 変数と異なるエントリを検索し、該当するエントリの Destination IP Address 項目の値を Target 変数に設定する(ステップ 8 9 0 4)。ステップ 8 9 0 4 の Target 変数を順番にすべて取得し、Target 変数の値が NULL 値に等しくないかチェックする(ステップ 8 9 0 5)。

ステップ 8 9 0 5 で Target 変数が NULL 値に等しくない場合は、P F テーブル 1 1 2 4 のすべてのエントリの中から Source IP Address 項目の値が IF 変数(Unit1 変数と Unit2 変数の内で、図 8 8 で IF と認識された機器)で Destination IP Address 項目の値が Target 変数に等しいエントリを検索し、該当するエントリの Source Port 項目の値を IFT 変数に設定する(ステップ 8 9 0 6)。

ステップ 8 9 0 5 で Target 変数が NULL 値に等しい場合は、Pport 変数に NULL 値

、Cport変数にNULL値を設定し(ステップ8909)、ステップ8901から繰り返す。

【0138】

ステップ8906で該当するエントリが存在し、IFT変数の値がNULLに等しくないかチェックし(ステップ8907)、ステップ8907でIFT変数の値がNULLに等しくない場合は、Pport変数にIFT変数の値、Cport変数にCFIF変数の値を設定し(ステップ8908)、ステップ8901から繰り返す。

ステップ8907でIFT変数の値がNULLに等しい場合は、ステップ8904から繰り返す。図46および図47のR-IF-CFモデルで示した接続検出条件に基づいて、図89のフローでは図28に示した方法で接続関係を検出する。

【0139】

図90はオートディスカバリモジュール1113によるTSテーブル1125の作成時のR-CF-IFモデルの接続条件チェック処理を示すフローチャートである。

オートディスカバリモジュール1113はR-CF-IFモデルの接続条件チェック処理要求を待ち(ステップ9001)、R-CF-IFモデルの接続条件チェック処理要求を受信すると(ステップ9002)、PFテーブル1124のすべてのエントリの中からSource IP Address項目の値がCF変数(Unit1変数とUnit2変数の内で、図88でCFと認識された機器)でDestination IP Address項目の値がIF変数(Unit1変数とUnit2変数の内で、図88でIFと認識された機器)に等しいエントリを検索し、該当するエントリのSource Port項目の値をCFIF変数に設定する(ステップ9003)。

【0140】

同様に、PFテーブル1124のすべてのエントリの中からSource IP Address項目の値がIF変数(Unit1変数とUnit2変数の内でIFと認識された機器)でDestination IP Address項目の値がRoot変数に等しいエントリを検索し、該当するエントリのSource Port項目の値をIFR変数に設定する(ステップ9004)。

最後に、Pport変数にCFIF変数の値、Cport変数にIFR変数の値を設定し(ステップ9005)、ステップ9001から繰り返す。図46及び図47のR-CF-IFモデル

ルでは接続検出条件がないことから、図 9 0 のフローでは図 2 4 に示した方法で接続関係を検出する。

【 0 1 4 1 】

図 9 1 はオートディスカバリモジュール 1 1 1 3 による T S テーブル 1 1 2 5 の作成時の集合 (R, CF, SF) の接続条件チェック処理を示すフローチャートである。

オートディスカバリモジュール 1 1 1 3 は集合 (R, CF, SF) の接続条件チェック処理要求を待ち (ステップ 9 1 0 1)、集合 (R, CF, SF) の接続条件チェック処理要求を受信すると (ステップ 9 1 0 2)、P F テーブル 1 1 2 4 のすべてのエントリの中から Source IP Address 項目の値が CF 変数 (Unit1 変数と Unit2 変数の内で CF と認識された機器) で Destination IP Address 項目の値が Root 変数に等しいエントリを検索し、該当するエントリの Source Port 項目の値を CFR 変数に設定する (ステップ 9 1 0 3)。

【 0 1 4 2 】

同様に、P F テーブル 1 1 2 4 のすべてのエントリの中から Source IP Address 項目の値が CF 変数 (Unit1 変数と Unit2 変数の内で CF と認識された機器) で Destination IP Address 項目の値が SF 変数 (Unit1 変数と Unit2 変数の内で SF と認識された機器) に等しいエントリを検索し、該当するエントリの Source Port 項目の値を CFSF 変数に設定する (ステップ 9 1 0 4)。CFR 変数の値と CFSF 値を比較し (ステップ 9 1 0 5)、CF から見て、R と SF が別ポートに接続しているかチェックする。

ステップ 9 1 0 5 で CFR 変数の値と CFSF の値が等しい場合は、Paddr 変数に SF 変数の値、Caddr 変数に CF 変数の値、Model 変数に R-SF-CF を設定し、図 9 2 に示す R-SF-CF モデルの接続検出条件チェック処理を実行し (ステップ 9 1 0 6)、ステップ 9 1 0 1 から繰り返す。

ステップ 9 1 0 5 で CFR 変数の値と CFSF の値が等しくない場合は、Paddr 変数に CF 変数の値、Caddr 変数に SF 変数の値、Model 変数に R-CF-SF を設定し、図 9 3 に示す R-CF-SF モデルの接続検出条件チェック処理を実行し (ステップ 9 1 0 7)、ステップ 9 1 0 1 から繰り返す。

【 0 1 4 3 】

図 9 2 はオートディスカバリモジュール 1 1 1 3 による T S テーブル 1 1 2 5 の作成時の R-SF-CF モデルの接続条件チェック処理を示すフローチャートである。

オートディスカバリモジュール 1 1 1 3 は R-SF-CF モデルの接続条件チェック処理要求を待ち(ステップ 9 2 0 1)、R-SF-CF モデルの接続条件チェック処理要求を受信すると(ステップ 9 2 0 2)、P F テーブル 1 1 2 4 のすべてのエントリの中から Source IP Address 項目の値が CF 変数 (Unit1 変数と Unit2 変数の内で、図 9 1 で CF と認識された機器) で Destination IP Address 項目の値が SF 変数 (Unit1 変数と Unit2 変数の内で、図 9 1 で SF と認識された機器) に等しいエントリを検索し、該当するエントリの Source Port 項目の値を CFSF 変数に設定する(ステップ 9 2 0 3)。

【 0 1 4 4 】

同様にして、P F テーブル 1 1 2 4 のすべてのエントリの中から Source IP Address 項目の値が CF 変数 (Unit1 変数と Unit2 変数の内で CF と認識された機器) で Source Port 項目の値が CFSF 変数と異なるエントリを検索し、該当するエントリの Destination IP Address 項目の値を Target 変数に設定する(ステップ 9 2 0 4)。

ステップ 9 2 0 4 の Target 変数を順番にすべて取得し、Target 変数の値が NULL 値に等しくないかチェックする(ステップ 9 2 0 5)。ステップ 9 2 0 5 で Target 変数が NULL 値に等しくない場合は、P F テーブル 1 1 2 4 におけるすべてのエントリの中から Source IP Address 項目の値が SF 変数 (Unit1 変数と Unit2 変数の内で、図 9 1 で SF と認識された機器) で Destination IP Address 項目の値が Target 変数に等しいエントリを検索し、該当するエントリの Source Port 項目の値を SFT 変数に設定する(ステップ 9 2 0 6)。

【 0 1 4 5 】

ステップ 9 2 0 5 で Target 変数が NULL 値に等しい場合は、Pport 変数に NULL 値、Cport 変数に NULL 値を設定し(ステップ 9 2 0 9)、ステップ 9 2 0 1 から繰り返す。ステップ 9 2 0 6 で該当するエントリが存在し、SFT 変数の値が NULL に等しくないかチェックし(ステップ 9 2 0 7)、ステップ 9 2 0 7 で SFT 変数の値が NULL に等しくない場合は、Pport 変数に SFT 変数の値、Cport 変数に CFSF 変数の値を

設定し(ステップ9 2 0 8)、ステップ9 2 0 1から繰り返す。

ステップ9 2 0 7でSFT変数の値がNULLに等しい場合は、ステップ9 2 0 4から繰り返す。図4 6及び図4 7のR-SF-CFモデルで示した接続検出条件に基づいて、図9 2では図3 4に示した方法で接続関係を検出する。

【0 1 4 6】

図9 3はオートディスカバリモジュール1 1 1 3によるTSテーブル1 1 2 5の作成時のR-CF-SFモデルの接続条件チェック処理を示すフローチャートである。

オートディスカバリモジュール1 1 1 3はR-CF-SFモデルの接続条件チェック処理要求を待ち(ステップ9 3 0 1)、R-CF-SFモデルの接続条件チェック処理要求を受信すると(ステップ9 3 0 2)、PFテーブル1 1 2 4のすべてのエントリの中からSource IP Address項目の値がCF変数(Unit1変数とUnit2変数の内で、図9 1でCFと認識された機器)でDestination IP Address項目の値がSF変数(Unit1変数とUnit2変数の内で、図9 1でSFと認識された機器)に等しいエントリを検索し、該当するエントリのSource Port項目の値をCFSF変数に設定する(ステップ9 3 0 3)。

【0 1 4 7】

同様にして、PFテーブル1 1 2 4におけるすべてのエントリの中からSource IP Address項目の値がCF変数(Unit1変数とUnit2変数の内でCFと認識された機器)でSource Port項目の値がCFSF変数と異なるエントリを検索し、該当するエントリのDestination IP Address項目の値をTarget変数に設定する(ステップ9 3 0 4)。

ステップ9 3 0 4のTarget変数を順番にすべて取得し、Target変数の値がNULL値に等しくないかチェックする(ステップ9 3 0 5)。ステップ9 3 0 5でTarget変数がNULL値に等しくない場合は、PFテーブル1 1 2 4のすべてのエントリの中からSource IP Address項目の値がSF変数(Unit1変数とUnit2変数の内で、図9 1でSFと認識された機器)でDestination IP Address項目の値がTarget変数に等しいエントリを検索し、該当するエントリのSource Port項目の値をSFT変数に設定する(ステップ9 3 0 6)。

【 0 1 4 8 】

ステップ 9 3 0 5 で Target 変数が NULL 値に等しい場合は、Pport 変数に NULL 値、Cport 変数に NULL 値を設定し(ステップ 9 3 0 9)、ステップ 9 3 0 1 から繰り返す。ステップ 9 3 0 6 で該当するエントリが存在し、SFT 変数の値が NULL に等しくないかチェックし(ステップ 9 3 0 7)、ステップ 9 3 0 7 で SFT 変数の値が NULL に等しくない場合は、Pport 変数に CFSF 変数の値、Cport 変数に SFT 変数の値を設定し(ステップ 9 3 0 8)、ステップ 9 3 0 1 から繰り返す。

ステップ 9 3 0 7 で SFT 変数の値が NULL に等しい場合は、ステップ 9 3 0 4 から繰り返す。図 4 6 及び図 4 7 の R-CF-SF モデルで示した接続検出条件に基づいて、図 9 3 では図 2 6 に示した方法で接続関係を検出する。

【 0 1 4 9 】

図 9 4 はオートディスカバリモジュール 1 1 1 3 による TS テーブル 1 1 2 5 の作成時の集合(R, IF, IF)の接続条件チェック処理を示すフローチャートである。

オートディスカバリモジュール 1 1 1 3 は集合(R, IF, IF)の接続条件チェック処理要求を待ち(ステップ 9 4 0 1)、集合(R, IF, IF)の接続条件チェック処理要求を受信すると(ステップ 9 4 0 2)、PF テーブル 1 1 2 4 のすべてのエントリの中から Source IP Address 項目の値が IF1 変数(Unit1 変数もしくは Unit2 変数のどちらか一方の機器)で Destination IP Address 項目の値が Root 変数に等しいエントリを検索し、該当するエントリの Source Port 項目の値を IF1R 変数に設定する(ステップ 9 4 0 3)。

【 0 1 5 0 】

同様に、PF テーブル 1 1 2 4 におけるすべてのエントリの中から Source IP Address 項目の値が IF2 変数(Unit1 変数もしくは Unit2 変数の内で IF1 とは異なる機器)で Destination IP Address 項目の値が Root 変数に等しいエントリを検索し、該当するエントリの Source Port 項目の値を IF2R 変数に設定する(ステップ 9 4 0 4)。

次に、図 9 5 に示す R-IF-IF モデルの接続検出条件チェック処理を実行し(ステップ 9 4 0 5)、接続ポートを決定する(IF1IF2(IF2IF1))。

IF1IF2変数の値がNULLに等しくない条件とIF2IF1変数の値がNULLに等しくない条件を同時に満たすかどうかをチェックし(ステップ9406)、IF1とIF2の接続ポートが発見できたかチェックする。ステップ9406でIF1IF2変数の値がNULLに等しくなく、かつIF2IF1変数の値がNULLに等しくない場合は、Paddr変数にIF1変数の値、Caddr変数にIF2変数の値、Pport変数にIF1IF2の値、Cport変数にIF2R(IF2IF1)の値、Model変数にR-IF-IFを設定し(ステップ9407)、ステップ9401から繰り返す。

【0151】

ステップ9406でIF1IF2変数の値がNULLに等しい、またはIF2IF1変数の値がNULLに等しい場合は、IF1R変数の値とIF2R変数の値、IF1変数の値とIF2変数の値を入れ替えて、図95に示すR-IF-IFモデルの接続検出条件チェック処理を実行し(ステップ9408)、接続ポートを決定する(IF1IF2(IF2IF1))。IF1IF2変数の値がNULLに等しくない条件とIF2IF1変数の値がNULLに等しくない条件を同時に満たすかどうかをチェックし(ステップ9409、IF1とIF2の接続ポートが発見できたかチェックする。ステップ9409でIF1IF2変数の値がNULLに等しくなく、かつIF2IF1変数の値がNULLに等しくない場合は、Paddr変数にIF1変数の値、Caddr変数にIF2変数の値、Pport変数にIF1IF2の値、Cport変数にIF2R(IF2IF1)の値、Model変数にR-IF-IFを設定し(ステップ9410)、ステップ9401から繰り返す。ただし、ステップ9410で設定するIF1変数とIF2変数の値はステップ9407で設定するIF1変数とIF2変数とは入れ替えた値である。ステップ9409でIF1IF2変数の値がNULLに等しい、またはIF2IF1変数の値がNULLに等しい場合はステップ9401から繰り返す。

【0152】

図95はオートディスカバリモジュール1113によるTSテーブル1125の作成時のR-IF-IFモデルの接続条件チェック処理を示すフローチャートである。

オートディスカバリモジュール1113はR-IF-IFモデルの接続条件チェック処理要求を待ち(ステップ9501)、R-IF-IFモデルの接続条件チェック処理要求を受信すると(ステップ9502)、PFテーブル1124のすべてのエントリ

の中からSource IP Address項目の値がIF1変数(Unit1変数とUnit2変数の内で、図9 4でIF1と認識された機器)でSource Port項目の値がIF1R変数(図9 4のIF1からRootへの接続ポート)に等しくないエントリを検索し、該当するエントリのDestination IP Address項目の値をTarget1変数に設定する(ステップ9 5 0 3)。

ステップ9 5 0 3のTarget1変数を順番にすべて取得し、Target1変数の値がNULL値に等しくないかチェックする(ステップ9 5 0 4)。ステップ9 5 0 4でTarget1変数がNULL値に等しくない場合は、PFテーブル1 1 2 4のすべてのエントリの中からSource IP Address項目の値がIF2変数(Unit1変数とUnit2変数の内で、図9 4でIF2と認識された機器)でDestination IP Address項目の値がTarget1変数に等しいエントリを検索し、該当するエントリのSource Port項目の値をIF2T1変数に設定する(ステップ9 5 0 5)。

【0 1 5 3】

ステップ9 5 0 4でTarget1変数がNULL値に等しい場合は、IF1IF2変数にNULL値、IF2IF1変数にNULL値を設定し(ステップ9 5 1 2)、ステップ9 5 0 1から繰り返す。

【0 1 5 4】

ステップ9 5 0 5で該当するエントリが存在し、IF2T1変数の値がNULLに等しくない条件とIF2T1変数の値がIF2R変数の値に等しい条件を同時に満たすかどうかをチェックし(ステップ9 5 0 6)、ステップ9 5 0 6でIF2T1変数の値がNULLに等しくなく、かつIF2T1変数の値がIF2R変数の値に等しい場合は、PFテーブル1 1 2 4のすべてのエントリの中からSource IP Address項目の値がIF2変数(Unit1変数とUnit2変数の内で、図9 4でIF2と認識された機器)でSource Port項目の値がIF2R変数(図9 4のIF2からRootへの接続ポート)に等しくないエントリを検索し、該当するエントリのDestination IP Address項目の値をTarget2変数に設定する(ステップ9 5 0 7)。

ステップ9 5 0 6でIF2T1変数の値がNULLに等しい、またはIF2T1変数の値がIF2R変数の値に等しくない場合は、ステップ9 5 0 3から繰り返す。ステップ9 5 0 7のTarget2変数を順番にすべて取得し、Target2変数の値がNULL値に等しくないかチェックする(ステップ9 5 0 8)。

【 0 1 5 5 】

ステップ 9 5 0 8 で Target2 変数が NULL 値に等しくない場合は、P F テーブル 1 1 2 4 のすべてのエントリの中から Source IP Address 項目の値が IF1 変数 (Unit1 変数と Unit2 変数の内で、図 9 4 で IF1 と認識された機器) で Destination IP Address 項目の値が Target2 変数に等しいエントリを検索し、該当するエントリの Source Port 項目の値を IF1T2 変数に設定する (ステップ 9 5 0 9)。

ステップ 9 5 0 8 で Target2 変数が NULL 値に等しい場合は、ステップ 9 5 0 3 から繰り返す。ステップ 9 5 0 9 で該当するエントリが存在し、IF1T2 変数の値が NULL に等しくない条件と IF1T2 変数の値が IF1R 変数の値に等しくない条件を同時に満たすかどうかをチェックし (ステップ 9 5 1 0)、ステップ 9 5 1 0 で IF1T2 変数の値が NULL に等しくなく、かつ IF1T2 変数の値が IF1R 変数の値に等しくない場合は、IF1IF2 変数に IF1T2 変数の値、IF2IF1 変数に IF2T1 変数の値を設定し (ステップ 9 5 1 1)、ステップ 9 5 0 1 から繰り返す。ステップ 9 5 1 0 で IF1T2 変数の値が NULL に等しい、または IF1T2 変数の値が IF2R 変数の値に等しい場合は、ステップ 9 5 0 7 から繰り返す。

図 4 6 及び図 4 7 の R-IF-IF モデルで示した接続検出条件に基づいて、図 9 5 では図 3 0 に示した方法で接続関係を検出する。

【 0 1 5 6 】

図 9 6 はオートディスカバリモジュール 1 1 1 3 による T S テーブル 1 1 2 5 の作成時の集合 (R, IF, SF) の接続条件チェック処理を示すフローチャートである。

オートディスカバリモジュール 1 1 1 3 は集合 (R, IF, SF) の接続条件チェック処理要求を待ち (ステップ 9 6 0 1)、集合 (R, IF, SF) の接続条件チェック処理要求を受信すると (ステップ 9 6 0 2)、P F テーブル 1 1 2 4 のすべてのエントリの中から Source IP Address 項目の値が IF 変数 (Unit1 変数と Unit2 変数の内で IF と認識された機器) で Destination IP Address 項目の値が Root 変数に等しいエントリを検索し、該当するエントリの Source Port 項目の値を IFR 変数に設定する (ステップ 9 6 0 3)。

【 0 1 5 7 】

図 9 7 に示す R-SF-IF モデルの接続検出条件チェック処理を実行し(ステップ 9 6 0 4)、ステップ 9 6 0 4 で設定される Paddr 変数の値が NULL 値に等しいかチェックする(ステップ 9 6 0 5)。ステップ 9 6 0 5 で Paddr 変数の値が NULL 値に等しい場合は、図 9 9 に示す R-IF-SF モデルの接続検出条件チェック処理を実行し(ステップ 9 6 0 6)、ステップ 9 6 0 1 から繰り返す。ステップ 9 6 0 5 で Paddr 変数の値が NULL 値に等しくない場合は、ステップ 9 6 0 1 から繰り返す。

【 0 1 5 8 】

図 9 7 及び図 9 8 は、オートディスカバリモジュール 1 1 1 3 による T S テーブル 1 1 2 5 の作成時の R-SF-IF モデルの接続条件チェック処理を示すフローチャートである。

オートディスカバリモジュール 1 1 1 3 は R-SF-IF モデルの接続条件チェック処理要求を待ち(ステップ 9 7 0 1)、R-SF-IF モデルの接続条件チェック処理要求を受信すると(ステップ 9 7 0 2)、P F テーブル 1 1 2 4 のすべてのエントリの中から Source IP Address 項目の値が IF 変数(Unit1 変数と Unit2 変数の内で、図 9 6 で IF と認識された機器)で Destination IP Address 項目の値が Root 変数に等しいエントリを検索し、該当するエントリの Source Port 項目の値を IFR 変数に設定する(ステップ 9 7 0 3)。

【 0 1 5 9 】

同様にして、P F テーブル 1 1 2 4 におけるすべてのエントリの中から Source IP Address 項目の値が IF 変数(Unit1 変数と Unit2 変数の内で、図 9 6 で IF と認識された機器)で Source Port 項目の値が IFR 変数に等しいエントリを 2 つ検索し、該当するエントリの Destination IP Address 項目の値をそれぞれ Target1 変数、Target2 変数、Source Port 項目の値を IFT1 変数、IFT2 変数に設定する(ステップ 9 7 0 4)。

【 0 1 6 0 】

ステップ 9 7 0 4 の Target1 変数と Target2 変数の組合わせを順番にすべて取得し、Target1 変数の値が NULL 値に等しくなく、かつ Target2 変数の値が NULL 値に等しくないかチェックする(ステップ 9 7 0 5)。ステップ 9 7 0 5 の条件を満たす場合は、P F テーブル 1 1 2 4 のすべてのエントリの中から Source IP Address

項目の値がSF変数(Unit1変数とUnit2変数の内で、図9 6でSFと認識された機器)でDestination IP Address項目の値がTarget1変数に等しいエントリを検索し、該当するエントリのSource Port項目の値をSFT1変数に設定する。

同様に、PFテーブル1 1 2 4におけるすべてのエントリの中からSource IP Address項目の値がSF変数(Unit1変数とUnit2変数の内で、図9 6でSFと認識された機器)でDestination IP Address項目の値がTarget2変数に等しいエントリを検索し、該当するエントリのSource Port項目の値をSFT2変数に設定する(ステップ9 7 0 6)。

ステップ9 7 0 5の条件を満たさない場合は、Paddr変数にNULL値、Caddr変数にNULL値、Pport変数にNULL値、Cport変数にNULL値、Model変数にR-SF-IFを設定し(ステップ9 7 1 3)、ステップ9 7 0 1から繰り返す。

【0 1 6 1】

ステップ9 7 0 6で該当するエントリが存在し、SFT1変数の値がNULLに等しくなく、かつSFT2変数の値がNULLに等しくなく、かつSFT1変数の値がSFT2変数の値に等しくないかチェックし(ステップ9 7 0 7)、ステップ9 7 0 7の条件を満たす場合は、PFテーブル1 1 2 4のすべてのエントリの中からSource IP Address項目の値がIF変数(Unit1変数とUnit2変数の内で、図9 6でIFと認識された機器)でSource Port項目の値がIFR変数(図9 6のIFからRootへの接続ポート)に等しくなく、かつIFT1変数に等しくなく、IFT2変数に等しくないエントリを検索し、該当するエントリのDestination IP Address項目の値をTarget3変数、Source Port項目の値をIFT3に設定する(ステップ9 7 0 8)。

【0 1 6 2】

ステップ9 7 0 7の条件を満たさない場合は、ステップ9 7 0 4から繰り返す。

ステップ9 7 0 8のTarget3変数を順番にすべて取得し、Target3変数の値がNULL値に等しくないかチェックする(ステップ9 7 0 9)。ステップ9 7 0 9でTarget3変数がNULL値に等しくない場合は、PFテーブル1 1 2 4のすべてのエントリの中からSource IP Address項目の値がSF変数(Unit1変数とUnit2変数の内で、図9 6でSFと認識された機器)でDestination IP Address項目の値がTarget3変数

に等しいエントリを検索し、該当するエントリのSource Port項目の値をSFT3変数に設定する(ステップ9710)。

【0163】

ステップ9709でTarget3変数がNULL値に等しい場合は、ステップ9704から繰り返す。ステップ9710で該当するエントリが存在するかチェックし(ステップ9711)、ステップ9711の条件を満たす場合は、Paddr変数にSF変数の値、Caddr変数にIF変数の値、Pport変数にSFT3変数の値、Cport変数にIFR変数の値(IFT1変数の値、IFT2変数の値)、Model = R-SF-IFを設定し(ステップ9712)、ステップ9701から繰り返す。

ステップ9711の条件を満たさない場合は、ステップ9708から繰り返す。図46及び図47のR-SF-IFモデルで示した接続検出条件に基づいて、図97及び図98では図36に示した方法で接続関係を検出する。

【0164】

図99及び図100は、オートディスカバリモジュール1113によるTSテーブル1125の作成時のR-IF-SFモデルの接続条件チェック処理を示すフローチャートである。

オートディスカバリモジュール1113はR-IF-SFモデルの接続条件チェック処理要求を待ち(ステップ9901)、R-IF-SFモデルの接続条件チェック処理要求を受信すると(ステップ9902)、PFテーブル1124のすべてのエントリの中からSource IP Address項目の値がIF変数(Unit1変数とUnit2変数の内で、図96でIFと認識された機器)でDestination IP Address項目の値がRoot変数に等しいエントリを検索し、該当するエントリのSource Port項目の値をIFR変数に設定する(ステップ9903)。

【0165】

同様に、PFテーブル1124のすべてのエントリの中からSource IP Address項目の値がIF変数(Unit1変数とUnit2変数の内で、図96でIFと認識された機器)でSource Port項目の値がIFR変数に等しくないエントリを2つ検索し、該当するエントリのDestination IP Address項目の値をそれぞれTarget1変数、Target2変数、Source Port項目の値をIFT1変数、IFT2変数に設定する(ステップ99

0 4)。

【 0 1 6 6 】

ステップ 9 9 0 4 の Target1 変数と Target2 変数の組合わせを順番にすべて取得し、Target1 変数の値が NULL 値に等しくなく、かつ Target2 変数の値が NULL 値に等しくないかチェックする (ステップ 9 9 0 5)。

ステップ 9 9 0 5 の条件を満たす場合は、P F テーブル 1 1 2 4 のすべてのエントリの中から Source IP Address 項目の値が SF 変数 (Unit1 変数と Unit2 変数の内で、図 9 6 で SF と認識された機器) で Destination IP Address 項目の値が Target1 変数に等しいエントリを検索し、該当するエントリの Source Port 項目の値を SFT1 変数に設定する。

【 0 1 6 7 】

同様に、P F テーブル 1 1 2 4 のすべてのエントリの中から Source IP Address 項目の値が SF 変数 (Unit1 変数と Unit2 変数の内で、図 9 6 で SF と認識された機器) で Destination IP Address 項目の値が Target2 変数に等しいエントリを検索し、該当するエントリの Source Port 項目の値を SFT2 変数に設定する (ステップ 9 9 0 6)。

【 0 1 6 8 】

ステップ 9 9 0 5 の条件を満たさない場合は、Paddr 変数に NULL 値、Caddr 変数に NULL 値、Pport 変数に NULL 値、Cport 変数に NULL 値、Model 変数に R-IF-SF を設定し (ステップ 9 9 1 3)、ステップ 9 9 0 1 から繰り返す。

ステップ 9 9 0 6 で該当するエントリが存在し、SFT1 変数の値が NULL に等しくなく、かつ SFT2 変数の値が NULL に等しくなく、かつ SFT1 変数の値が SFT2 変数の値に等しいかチェックし (ステップ 9 9 0 7)、ステップ 9 9 0 7 の条件を満たす場合は、P F テーブル 1 1 2 4 のすべてのエントリの中から Source IP Address 項目の値が IF 変数 (Unit1 変数と Unit2 変数の内で、図 9 6 で IF と認識された機器) で Source Port 項目の値が IFR 変数 (図 9 6 の IF から Root への接続ポート) に等しくなく、かつ IFT1 変数に等しくなく、IFT2 変数に等しくないエントリを検索し、該当するエントリの Destination IP Address 項目の値を Target3 変数、Source Port 項目の値を IFT3 に設定する (ステップ 9 9 0 8)。ステップ 9 9 0 7 の条件を満たさ

ない場合は、ステップ 9 8 0 4 から繰り返す。

【 0 1 6 9 】

ステップ 9 9 0 8 の Target3 変数を順番にすべて取得し、Target3 変数の値が NULL 値に等しくないかチェックする (ステップ 9 9 0 9)。ステップ 9 9 0 9 で Target3 変数が NULL 値に等しくない場合は、P F テーブル 1 1 2 4 のすべてのエントリの中から Source IP Address 項目の値が SF 変数 (Unit1 変数と Unit2 変数の内で、図 9 6 で SF と認識された機器) で Destination IP Address 項目の値が Target3 変数に等しいエントリを検索し、該当するエントリの Source Port 項目の値を SFT3 変数に設定する (ステップ 9 9 1 0)。ステップ 9 9 0 9 で Target3 変数が NULL 値に等しい場合は、ステップ 9 9 0 4 から繰り返す。

【 0 1 7 0 】

ステップ 9 9 1 0 で該当するエントリが存在し、SFT3 変数の値が NULL に等しくなく、かつ SFT3 変数の値が SFT1 変数の値に等しくなく、かつ SFT3 変数の値が SFT2 変数の値に等しくないかチェックし (ステップ 9 9 1 1)、ステップ 9 9 1 1 の条件を満たす場合は、Paddr 変数に IF 変数の値、Caddr 変数に SF 変数の値、Pport 変数に IFT1 変数の値 (IFT1 変数の値、IFT2 変数の値)、Cport 変数に SFT3 変数の値、Model = R-IF-SF を設定し (ステップ 9 9 1 2)、ステップ 9 9 0 1 から繰り返す。

ステップ 9 9 1 1 の条件を満たさない場合は、ステップ 9 9 0 8 から繰り返す。図 4 6 及び図 4 7 の R-IF-SF モデルで示した接続検出条件に基づいて、図 9 9 及び図 1 0 0 では図 3 2 に示した方法で接続関係を検出する。

【 0 1 7 1 】

図 1 0 1 は、オートディスカバリモジュール 1 1 1 3 による T S テーブル 1 1 2 5 の作成時の集合 (R, SF, SF) の接続条件チェック処理を示すフローチャートである。

オートディスカバリモジュール 1 1 1 3 は集合 (R, SF, SF) の接続条件チェック処理要求を待ち (ステップ 1 0 1 0 1)、集合 (R, SF, SF) の接続条件チェック処理要求を受信すると (ステップ 1 0 1 0 2)、Paddr 変数に NULL 値、Caddr 変数に NULL 値、Pport 変数に NULL 値、Cport 変数に NULL 値、Model = R-SF-SF を設定し (ステップ 1 1 0 0 3)、ステップ 1 1 0 0 1 から繰り返す。図 4 6 及び図 4 7 の R-S

F-SFモデルで任意の条件で接続検出が不可能であるため、図101では図38に示したように接続関係の検出を中止する。

【0172】

図102はオートディスカバリモジュール1113によるTSテーブル1125の作成時のTSテーブルに対するエントリ追加処理を示すフローチャートである。

オートディスカバリモジュール1113はTSテーブル1125に対するエントリ追加処理要求を待ち(ステップ10201)、TSテーブルに対するエントリ追加処理要求を受信すると(ステップ10202)、Paddr変数がNULL値に等しく、かつCaddr変数がNULL値に等しく、かつPport変数がNULL値に等しく、かつCport変数がNULL値に等しいかチェックし(ステップ10203)、ステップ10203の条件を満たす場合は、図104に示す親子関係と接続関係が不明なネットワーク中継装置のエントリ追加処理を実行し(ステップ10204)、ステップ10201から繰り返す。

【0173】

ステップ10203の条件を満たさない場合は、Model変数がR-SF-CFに等しい、もしくはR-SF-IFに等しいかチェックし(ステップ10205)、ステップ10205の条件を満たす場合は図105に示す親子関係だけが不明なネットワーク中継装置のエントリ追加処理を実行し(ステップ10206)、ステップ10201から繰り返す。

ステップ10205の条件を満たさない場合は、図106に示す親子関係と接続関係が自明なネットワーク中継装置のエントリ追加処理を実行し(ステップ10207)、ステップ10201から繰り返す。

【0174】

図103はオートディスカバリモジュール1113によるTSテーブル1125の作成時のTSテーブル1125に対するRootエントリ追加処理を示すフローチャートである。

オートディスカバリモジュール1113はTSテーブル1125に対するRootエントリ追加処理要求を待ち(ステップ10301)、TSテーブル1125に対

するRootエントリ追加処理要求を受信すると(ステップ10302)、ATテーブル1122を利用してRoot変数のIPアドレスからMacアドレスを解決し、Rphysaddr変数に設定する(ステップ10303)。最後に、TSテーブル1125に、Terminal IP Address項目がRoot変数の値、Terminal Mac Address項目がRphysaddr変数の値、Terminal Port項目がNULL値、Parent IP Address項目がNULL値、Parent Mac Address項目がNULL値、Parent Port項目がNULL値のエントリを追加し(ステップ10304)、ステップ10301から繰り返す。

【0175】

図104はオートディスカバリモジュール1113によるTSテーブル1125の作成時の親子関係と接続関係が不明なネットワーク中継装置のエントリ追加処理を示すフローチャートである。

オートディスカバリモジュール1113は親子関係と接続関係が不明なネットワーク中継装置のエントリ追加処理要求を待ち(ステップ10401)、親子関係と接続関係が不明なネットワーク中継装置のエントリ追加処理要求を受信すると(ステップ10402)、ATテーブル1122を利用してUnit1変数とUnit2変数(図83(図102)のUnit1変数、Unit2変数)のIPアドレスからMacアドレスを解決し、それぞれU1physaddr変数、U2physaddr変数に設定する(ステップ10403)。最後に、TSテーブル1125に、Terminal IP Address項目がUnit1変数の値、Terminal Mac Address項目がU1physaddr変数の値、Terminal Port項目がNULL値、Parent IP Address項目がNULL値、Parent Mac Address項目がNULL値、Parent Port項目がNULL値のエントリを追加し(ステップ10404)、Terminal IP Address項目がUnit2変数の値、Terminal Mac Address項目がU2physaddr変数の値、Terminal Port項目がNULL値、Parent IP Address項目がNULL値、Parent Mac Address項目がNULL値、Parent Port項目がNULL値のエントリを追加し(ステップ10405)、ステップ10401から繰り返す。

【0176】

図105はオートディスカバリモジュール1113によるTSテーブル1125の作成時の親子関係だけが不明なネットワーク中継装置のエントリ追加処理を示すフローチャートである。

オートディスカバリモジュール 1 1 1 3 は親子関係だけが不明なネットワーク中継装置のエントリ追加処理要求を待ち(ステップ 1 0 5 0 1)、親子関係だけが不明なネットワーク中継装置のエントリ追加処理要求を受信すると(ステップ 1 0 5 0 2)、A T テーブル 1 1 2 2 を利用して Paddr 変数と Caddr 変数(図 8 7 ~ 図 1 0 0 Paddr 変数、Caddr 変数)の I P アドレスから M a c アドレスを解決し、それぞれ Pphysaddr 変数、Cphysaddr 変数に設定する(ステップ 1 0 5 0 3)。

最後に T S テーブル 1 1 2 5 に、Terminal IP Address 項目が Paddr 変数の値、Terminal Mac Address 項目が Pphysaddr 変数の値、Terminal Port 項目が Pport 変数の値、Parent IP Address 項目が Caddr 変数の値、Parent Mac Address 項目が Cphysaddr 変数の値、Parent Port 項目が Cport 変数の値のエントリを追加し(ステップ 1 0 5 0 4)、Terminal IP Address 項目が Caddr 変数の値、Terminal Mac Address 項目が Cphysaddr 変数の値、Terminal Port 項目が Cport 変数の値、Parent IP Address 項目が Paddr 変数の値、Parent Mac Address 項目が Pphysaddr 変数の値、Parent Port 項目が Pport 変数の値のエントリを追加し(ステップ 1 0 5 0 5)、ステップ 1 0 5 0 1 から繰り返す。

【 0 1 7 7 】

図 1 0 6 はオートディスカバリモジュール 1 1 1 3 による T S テーブル 1 1 2 5 の作成時の親子関係と接続関係が自明なネットワーク中継装置のエントリ追加処理を示すフローチャートである。

オートディスカバリモジュール 1 1 1 3 は親子関係と接続関係が自明なネットワーク中継装置のエントリ追加処理要求を待ち(ステップ 1 0 6 0 1)、親子関係と接続関係が自明なネットワーク中継装置のエントリ追加処理要求を受信すると(ステップ 1 0 6 0 2)、A T テーブル 1 1 2 2 を利用して Paddr 変数と Caddr 変数(図 8 7 ~ 図 1 0 0 の Paddr 変数、Caddr 変数)の I P アドレスから M a c アドレスを解決し、それぞれ Pphysaddr 変数、Cphysaddr 変数に設定する(ステップ 1 0 6 0 3)。

最後に T S テーブル 1 1 2 5 に、Terminal IP Address 項目が Caddr 変数の値、Terminal Mac Address 項目が Cphysaddr 変数の値、Terminal Port 項目が Cport 変数の値、Parent IP Address 項目が Paddr 変数の値、Parent Mac Address 項目が Pp

hysaddr変数の値、Parent Port項目がPport変数の値のエントリを追加し(ステップ10604)、ステップ10601から繰り返す。

【0178】

図107及び図108は、オートディスカバリモジュール1113によるTSテーブル1125の作成時の親子関係の決定処理を示すフローチャートである。

オートディスカバリモジュール1113は親子関係の決定処理要求を待ち(ステップ10701)、親子関係の決定処理要求を受信すると(ステップ10702)、TSテーブル1125のすべてのエントリの中で、Parent IP Address項目の値が共通であるエントリが2つ存在するかチェックし、2エントリ存在する場合は、Terminal IP Address項目の値をそれぞれChild1変数、Child2変数、Parent IP Address項目の値をParent変数に設定する。今度は、Parent IP Address項目の値がChild1変数の値に等しく、Terminal IP Address項目の値がParent変数の値に等しいエントリが存在し、かつParent IP Address項目の値がChild2変数の値に等しく、Terminal IP Address項目の値がParent変数の値に等しいエントリが存在するかチェックする(ステップ10703)。

【0179】

ステップ10703の条件を満たす場合は、TSテーブル1125のすべてのエントリの中で、Terminal IP Address項目の値がChild1変数の値に等しく、かつParent IP Address項目の値がChild2変数に等しいエントリとTerminal IP Address項目の値がChild2変数の値に等しく、かつParent IP Address項目の値がChild1変数に等しいエントリが両方存在するかチェックする(ステップ10704)。

【0180】

ステップ10703の条件を満たさない場合は、図111に示す接続関係が不明なネットワーク中継装置の決定処理を実行し(ステップ10708)、図112に示すRootとネットワーク中継装置の親子関係決定処理を実行し(ステップ10709)、ステップ10701から繰り返す。

ステップ10704の条件を満たす場合は、図109に示す複数モデルの組合わせ処理を実行し(ステップ10705)、ステップ10703から繰り返す。

ステップ 1 0 7 0 4 の条件を満たさない場合は、T S テーブル 1 1 2 5 のすべてのエントリの中で、Terminal IP Address 項目の値が Child1 変数の値に等しく、かつ Parent IP Address 項目の値が Child2 変数に等しいエントリと Terminal IP Address 項目の値が Child2 変数の値に等しく、かつ Parent IP Address 項目の値が Child1 変数に等しいエントリのどちらか一方だけが存在するかチェックし(ステップ 1 0 7 0 6)、ステップ 1 0 7 0 6 の条件を満たす場合は、図 1 1 0 に示す T S テーブルエントリ結合処理を実行し(ステップ 1 0 7 0 7)、ステップ 1 0 7 0 3 から繰り返す。ステップ 1 0 7 0 6 の条件を満たさない場合は、ステップ 1 0 7 0 3 から繰り返す。

【 0 1 8 1 】

図 1 0 9 はオートディスカバリモジュール 1 1 1 3 による T S テーブル 1 1 2 5 の作成時の複数モデルの組合わせ処理を示すフローチャートである。

オートディスカバリモジュール 1 1 1 3 は複数モデルの組合わせ処理要求を待ち(ステップ 1 0 9 0 1)、複数モデルの組合わせ処理要求を受信すると(ステップ 1 0 9 0 2)、T S テーブル 1 1 2 5 のすべてのエントリの中で、Terminal IP Address 項目の値が Child1 変数(図 1 0 7 の Child1 変数)に等しく、かつ Parent IP Address 項目の値が Parent 変数(図 1 0 7 の Parent 変数)に等しいエントリを検索し、該当するエントリの Terminal Port 項目の値を C1Pport 変数に設定し、Parent Port 項目の値を PC1port 変数に設定する(ステップ 1 0 9 0 3)。

【 0 1 8 2 】

同様に、T S テーブル 1 1 2 5 のすべてのエントリの中で、Terminal IP Address 項目の値が Child2 変数(図 1 0 7 の Child2 変数)に等しく、かつ Parent IP Address 項目の値が Parent 変数(図 1 0 7 の Parent 変数)に等しいエントリを検索し、該当するエントリの Terminal Port 項目の値を C2Pport 変数に設定し、Parent Port 項目の値を PC2port 変数に設定する(ステップ 1 0 9 0 4)。

【 0 1 8 3 】

同様に、T S テーブル 1 1 2 5 のすべてのエントリの中で、Terminal IP Address 項目の値が Child1 変数(図 1 0 7 の Child1 変数)に等しく、かつ Parent IP Address 項目の値が Child2 変数(図 1 0 7 の Child2 変数)に等しいエントリを検索

し、該当するエントリのTerminal Port項目の値をC1C2port変数に設定し、Parent Port項目の値をC2C1port変数に設定する(ステップ10905)。

次に、C1Pport変数の値とC1C2port変数の値が等しいかチェック(C2Pport変数の値とC2C1port変数を比較することも可能)し(ステップ10906)、Child1から見て、ParentとChild2が等しいポートに接続しているかをチェックする。

ステップ10906の条件を満たす場合は、TSテーブル1125からTerminal IP AddressがChild2変数の値に等しく、Parent IP Address項目の値がChild1変数の値に等しいエントリを削除し(ステップ10907)、ステップ10901から繰り返す。

【0184】

ステップ10906の条件を満たさない場合は、TSテーブル1125からTerminal IP AddressがChild1変数の値に等しく、Parent IP Address項目の値がChild2変数の値に等しいエントリを削除し(ステップ10908)、ステップ10901から繰り返す。

図106のフローでは図55で示した方法で親子関係が不明なエントリの親子関係を検出する(Child1とChild2が格納されている2つのエントリの中で不必要なエントリを削除する)。

【0185】

図110はオートディスカバリモジュール1113によるTSテーブル1125の作成時のTSテーブル1125の結合処理を示すフローチャートである。

オートディスカバリモジュール1113はTSテーブルの結合処理要求を待ち(ステップ11001)、TSテーブルの結合処理要求を受信すると(ステップ11002)、TSテーブル1125のすべてのエントリの中で、Terminal IP Address項目の値がChild1変数(図107のChild1変数)に等しく、かつParent IP Address項目の値がChild2変数(図107のChild2変数)に等しいエントリが存在するかチェックし(ステップ11003)、ステップ11003の条件を満たす場合には、TSテーブル1125からTerminal IP AddressがChild1変数の値に等しく、Parent IP Address項目の値がParent変数(図107のParent変数)の値に等しいエントリを削除し(ステップ11004)、ステップ11001から繰り返す。

【 0 1 8 6 】

ステップ 1 1 0 0 3 の条件を満たさない場合には、TS テーブル 1 1 2 5 から Terminal IP Address が Child2 変数の値に等しく、Parent IP Address 項目の値が Parent 変数 (図 1 0 7 の Parent 変数) の値に等しいエントリを削除し (ステップ 1 1 0 0 5)、ステップ 1 1 0 0 1 から繰り返す。

【 0 1 8 7 】

図 1 1 1 はオートディスカバリモジュール 1 1 1 3 による TS テーブル 1 1 2 5 の作成時の接続関係が不明なネットワーク中継装置の決定処理を示すフローチャートである。

オートディスカバリモジュール 1 1 1 3 は接続関係が不明なネットワーク中継装置の決定処理要求を待ち (ステップ 1 1 1 0 1)、接続関係が不明なネットワーク中継装置の決定処理要求を受信すると (ステップ 1 1 1 0 2)、Units リスト変数 (図 8 3 (図 1 0 7) の Units リスト変数) のすべてのエントリを順番に取得し、Unit 変数に項目の値を設定し、未探索の Unit 変数があるかチェックする (ステップ 1 1 1 0 3)。

ステップ 1 1 1 0 3 で未検索の Unit 変数が存在する場合は、TS テーブル 1 1 2 5 から Terminal IP Address 項目が Unit 変数の値に等しいエントリを検索する (ステップ 1 1 1 0 4)。

ステップ 1 1 1 0 3 で未検索の Unit 変数が存在しない場合は、ステップ 1 1 1 0 1 から繰り返す。ステップ 1 1 1 0 4 で検索したエントリが存在するかチェックし (ステップ 1 1 1 0 5)、ステップ 1 1 1 0 5 で該当するエントリが存在する場合は、Parent IP Address 項目が NULL 値に等しいエントリだけが存在しているのかをチェックする (ステップ 1 1 1 0 7)。

【 0 1 8 8 】

ステップ 1 1 1 0 5 で該当するエントリが存在しない場合は、Paddr 変数、Caddr 変数に Unit 変数の値を設定し、図 1 0 6 に示した親子関係と接続関係が自明なネットワーク中継装置のエントリ追加処理を実行し (ステップ 1 1 1 0 6)、ステップ 1 1 1 0 3 から繰り返す。

ステップ 1 1 1 0 7 で Parent IP Address 項目が NULL 値に等しいエントリだけ

が存在する場合は、ステップ 1 1 1 0 3 から繰り返す。

ステップ 1 1 1 0 7 で Parent IP Address 項目が NULL 値に等しいエントリだけが含まれている場合は、TS テーブル 1 1 2 5 から Terminal IP Address 項目が Unit 変数に等しいエントリを削除し (ステップ 1 1 1 0 8)、ステップ 1 1 1 0 3 から繰り返す。

【 0 1 8 9 】

図 1 1 2 はオートディスカバリモジュール 1 1 1 3 による TS テーブル 1 1 2 5 の作成時の Root とネットワーク中継装置の親子関係決定処理を示すフローチャートである。

オートディスカバリモジュール 1 1 1 3 は Root とネットワーク中継装置の親子関係決定処理要求を待ち (ステップ 1 1 2 0 1)、Root とネットワーク中継装置の親子関係決定処理要求を受信すると (ステップ 1 1 2 0 2)、Units リスト変数 (図 8 3 (図 1 0 7) の Units リスト変数) のすべてのエントリを順番に取得し、Unit 変数に項目の値を設定し、未探索の Unit 変数があるかチェックする (ステップ 1 1 2 0 3)。

ステップ 1 1 2 0 3 で未探索の Unit 変数が存在する場合は、TS テーブル 1 1 2 5 から Terminal IP Address 項目が Unit 変数の値に等しいエントリを検索する (ステップ 1 1 2 0 4)。

【 0 1 9 0 】

ステップ 1 1 2 0 3 で未探索の Unit 変数が存在しない場合は、ステップ 1 1 2 0 1 から繰り返す。ステップ 1 1 2 0 4 で該当するエントリが存在する場合は、Parent IP Address 項目が NULL 値に等しいエントリだけが存在しているのかをチェックする (ステップ 1 1 2 0 5)。ステップ 1 1 2 0 5 で Parent IP Address 項目が NULL 値に等しいエントリだけが存在する場合は、図 1 1 3 に示す Root とネットワーク中継装置の接続ポート決定処理を実行し (ステップ 1 1 2 0 6)、Cport 変数と Pport 変数に値を設定する。最後に、TS テーブル 1 1 2 5 の該当するエントリの Terminal IP Address 項目に Unit 変数の値、Parent IP Address 項目に NULL 値、Terminal Port 項目に Cport 変数の値、Parent Port 項目に Pport 変数の値を設定し、エントリを更新した後、ステップ 1 1 2 0 3 から繰り返す。

【 0 1 9 1 】

図 1 1 3 はオートディスカバリモジュール 1 1 1 3 による T S テーブル 1 1 2 5 の作成時の Root とネットワーク中継装置の接続ポート決定処理を示すフローチャートである。

【 0 1 9 2 】

オートディスカバリモジュール 1 1 1 3 は Root とネットワーク中継装置の接続ポート決定処理要求を待ち(ステップ 1 1 3 0 1)、Root とネットワーク中継装置の接続ポート決定処理要求を受信すると(ステップ 1 1 3 0 2)、Unit 変数(図 1 1 2 の Unit 変数)を引数に図 8 5 に示したネットワーク機器の分類処理を実行し(ステップ 1 1 3 0 3)、Unit 変数の機器の分類を Category 変数に設定する。

次に、Category 変数が SF に等しいかチェックし(ステップ 1 1 3 0 4)、ステップ 1 1 3 0 4 の Category 変数の値が SF に等しい場合は、P F テーブル 1 1 2 4 から Source IP Address 項目が Unit 変数の値に等しく、Destination IP Address 項目の値が探索中のネットワークセグメントの I P アドレスに等しいエントリを検索し、該当するエントリの Source Port 項目の値を Cport 変数に設定する(ステップ 1 1 3 0 5)。

【 0 1 9 3 】

ステップ 1 1 3 0 4 の Category 変数の値が SF に等しくない場合は、P F テーブル 1 1 2 4 から Source IP Address 項目が Unit 変数の値に等しく、Destination I P Address 項目の値が Root 変数の値に等しいエントリを検索し、該当するエントリの Source Port 項目の値を Cport 変数に設定する(ステップ 1 1 3 0 6)。ステップ 1 1 3 0 5、ステップ 1 1 3 0 6 のステップが終了後、P F テーブル 1 1 2 4 から Source IP Address 項目が Root 変数の値に等しく、Destination IP Address 項目が Unit 変数の値(もしくは探索中のネットワークセグメントの任意の I P アドレス)に等しいエントリを検索し、該当するエントリの Source Port 項目の値を Pport 変数に設定し、ステップ 1 1 3 0 1 から繰り返す。

図 4 6 及び図 4 7 の R-CF モデル、R-IF モデル、R-SF モデルで示した接続検出条件に基づいて、図 1 1 3 のフローでは図 4 0、図 4 2、図 4 4 に示した方法で接続関係を検出する。

【 0 1 9 4 】

図 1 1 4 はオートディスカバリモジュール 1 1 1 3 による T S テーブル 1 1 2 5 の作成時のネットワーク中継装置と端末装置の接続の決定処理を示すフローチャートである。

オートディスカバリモジュール 1 1 1 3 はネットワーク中継装置と端末装置の接続の決定処理要求を待ち(ステップ 1 1 4 0 1)、ネットワーク中継装置と端末装置の接続の決定処理要求を受信すると(ステップ 1 1 4 0 2)、Units リスト変数(図 8 3 (図 1 0 7)の Units リスト変数)のすべてのエントリを順番に取得し、Parent 変数に項目の値を設定し、未探索の Parent 変数があるかチェックする(ステップ 1 1 4 0 3)。

ステップ 1 1 4 0 3 で未探索の Parent 変数が存在する場合は、T S テーブル 1 1 2 5 から Terminal IP Address 項目が Parent 変数の値に等しいエントリ、もしくは Parent IP Address 項目が Parent 変数の値に等しいエントリを検索し、Ports リスト変数に該当するエントリの Terminal Port 項目の値(Terminal IP Address 項目が Parent 変数の値に等しい場合)、もしくは Parent Port 項目の値(Parent IP Address 項目が Parent 変数の値に等しい場合)をすべて追加する(ステップ 1 1 4 0 4)。

【 0 1 9 5 】

ステップ 1 1 4 0 3 で未探索の Parent 変数が存在しない場合は、ステップ 1 1 4 0 1 から繰り返す。ステップ 1 1 4 0 4 の終了後、P F テーブル 1 1 2 4 から Source IP Address 項目が Parent 変数に等しく、Source Port 項目が Port リスト変数に含まれるポート番号をに等しくないエントリを検索し、該当するエントリの Destination IP Address 項目を Child 変数、Destination Port 項目の値を Cport 変数に設定する(ステップ 1 1 4 0 5)。

【 0 1 9 6 】

次に、A T テーブル 1 1 2 2 を利用して Parent 変数と Child 変数の I P アドレスから M a c アドレスに変換し、それぞれ Pphysaddr 変数、Cphysaddr 変数に設定し(ステップ 1 1 4 0 6)、T S テーブル 1 1 2 5 に Terminal IP Address 項目が Child 変数の値、Terminal Mac Address 項目が Cphysaddr 変数の値、Terminal Port

項目がCport変数の値、Parent IP Address項目がParent変数の値、Parent Mac Address項目がPphysaddr変数の値、Parent Port項目がPport変数の値のエントリを追加(すでに同一のエントリが存在する場合は何もしない)し(ステップ11407)、ステップ11403から繰り返す。

【0197】

図115はオートディスカバリモジュール1113がTSテーブル1125の作成時にインタフェースMIBを評価する処理を示すフローチャートである。

オートディスカバリモジュール1113はインタフェースMIB評価処理要求を待ち(ステップ11501)、インタフェースMIB評価処理要求としてRoot装置のIPアドレスの値(root変数に設定)を受信すると(ステップ11502)、TSテーブル1125のTerminal Port項目をキーに指定してTerminal Port項目がヌル値であるエントリを検索し、ヒットしたエントリのTerminal IP Address項目の値をTerminal変数、Parent IP Address項目の値をparent変数に設定し、Parent Port項目の値をPport変数に設定する(ステップ11503)。

【0198】

次に、TIテーブル1123のIP Address項目をキーに指定してTerminal変数に等しいエントリを検索し(ステップ11504)、ヒットしたエントリのMIB2項目の値が“1”(True)かチェックする(ステップ11505)。

MIB2項目の値が“0”(False)の場合はステップ11503から処理を繰り返す。MIB2項目の値が“1”(True)の場合はIPアドレスがParent変数の機器でポート番号がPport変数に等しいポートに関して、オブジェクト名としてifInOctets(ifOutOctets)を指定し、図64に示したフローでSNMP Get-Requestメッセージの送受信処理を実行し、統計分布を取得してstatisticsP変数に設定する。

【0199】

同様に、IPアドレスがTerminal変数の機器の全てのポート番号に関して、オブジェクト名としてifOutOctets(ifInOctets)を指定し、図64に示したフローでSNMP Get-Requestメッセージの送受信処理を実行し、統計分布を取得してstatisticsT[ポート番号]変数に設定する(ステップ11506)。

ステップ 1 1 5 0 6 の終了後、statisticsP変数とstatisticsT[ポート番号]変数との間に有意な差がないポート番号が存在するかチェックし(ステップ 1 1 5 0 7)、有意な差がないポート番号がない場合はステップ 1 1 5 0 3 から処理を繰り返す。ここで有意な差とは、例えば 2 つの値の差に一定の閾値を設けておき、2 つの値の差が閾値を超えるような場合には 2 つの値は異なると判断することを意味する。有意な差がないポート番号がある場合は T S テーブル 1 1 2 5 の該当するエントリの Terminal Port 項目に該当するポート番号を設定して T S テーブル 1 1 2 5 のエントリを更新する(ステップ 1 1 5 0 8)。ステップ 1 1 5 0 8 の終了後、ステップ 1 1 5 0 3 から処理を繰り返す。

【 0 2 0 0 】

図 1 1 6 は、図面表示プログラム 1 1 0 4 がネットワーク構成図面を表示する処理を示すフローチャートである。

図面表示プログラム 1 1 0 4 は、ネットワーク構成図面表示要求を待ち(ステップ 1 1 6 0 1)、ネットワーク構成図面表示要求を受信すると(ステップ 1 1 6 0 2)、図 6 9 に示したオートディスカバリモジュール 1 1 1 3 の A T テーブル 1 1 2 2 の作成要求処理を実行する(ステップ 1 1 6 0 3)。

次に、図 7 0 に示したオートディスカバリモジュール 1 1 1 3 の T I テーブル 1 1 2 3 の作成要求処理を実行する(ステップ 1 1 6 0 4)。次に、図 7 3 に示したオートディスカバリモジュール 1 1 1 3 の P F テーブル 1 1 2 4 の作成要求処理を実行する(ステップ 1 1 6 0 5)。

次に、図 8 1 に示したオートディスカバリモジュール 1 1 1 3 の T S テーブル 1 1 2 5 の作成要求処理を実行する(ステップ 1 1 6 0 6)。最後に、図 1 1 7 に示す描画処理を実行し(ステップ 1 1 6 0 7)、ステップ 1 1 6 0 1 から処理を繰り返す。

【 0 2 0 1 】

図 1 1 7 および図 1 1 8 は、図面表示プログラム 1 1 0 4 がネットワーク構成図面表示処理時に画面に描画する処理を示すフローチャートである。

図面表示プログラム 1 1 0 4 は描画処理要求を待ち(ステップ 1 1 7 0 1)、描画処理要求を受信すると(ステップ 1 1 7 0 2)、T S テーブル 1 1 2 5 すべての

エントリの探索を開始し、T S テーブル 1 1 2 5 に未探索のエントリがあるかチェックし、未探索のエントリがない場合はステップ 1 1 7 0 1 から処理を繰り返し、未探索のエントリがある場合はParent変数に該当するエントリのParent IP Address項目の値を、Child変数にTerminal IP Address項目の値を設定する(ステップ 1 1 7 0 3)。

次にParent変数の値がNULL値に等しいかどうかチェックし(ステップ 1 1 7 0 4)、Parent変数の値がNULL値に等しい場合はユーザにChild変数の機器の接続関係が検出できないことを通知し(ステップ 1 1 7 0 5)、ステップ 1 1 7 0 3 から繰り返す。

【 0 2 0 2 】

Parent変数の値がNULL値に等しくない場合は、T S テーブル 1 1 2 5 のすべてのエントリの中でParent IP Address項目の値がChild変数に等しく、Terminal IP Address項目の値がParent変数の値に等しいエントリが存在するかチェックし、Pport変数にParent Port項目の値を設定する(ステップ 1 1 7 0 6)。

該当するエントリが存在する場合はユーザにChild変数の機器の親子関係が検出できないことを通知し(ステップ 1 1 7 0 7)、ステップ 1 1 7 0 3 から繰り返す。該当するエントリがない場合は、Parent変数とChild変数の値をキーに指定し、T I テーブル 1 1 2 3 のIP Address項目を検索し、ヒットしたエントリのparent変数に対するHost Name項目の値を画面上に描画する(ステップ 1 1 7 0 8)。

ただし、Parentが既に描画されている場合やParent変数がヌル値の場合は未処理とする。

【 0 2 0 3 】

ステップ 1 1 7 0 8 の終了後、図 1 1 9 に示すノンインテリジェントハブ予測処理を実行し(ステップ 1 1 7 0 9)、Parent変数の機器とChild変数の機器の間にノンインテリジェントハブが稼動しているかチェックする(ステップ 1 1 7 0 9)。

ノンインテリジェントハブ予測処理の戻り値が“1”(True)の場合は、ノンインテリジェントハブをParent変数の機器の直下に描画し、ヒットしたエントリの

Child変数の機器に対するHost Name項目の値をノンインテリジェントハブの直下に描画する(ステップ11710)。ただし、ノンインテリジェントハブがParent変数の機器の直下に既に描画されている場合は未処理とする。

ノンインテリジェントハブ予測処理の戻り値が“0”(False)の場合は、ヒットしたエントリのChild変数の機器に対するHost Name項目の値をParentの直下に描画する(ステップ11711)。ステップ11710、ステップ11711の任意のステップが終了後、ステップ11703から処理を繰り返す。

ノンインテリジェントハブ予測処理はノンインテリジェントハブが稼動していることを認識することを目的としており、ノンインテリジェントハブの階層構造や段数は予測できないため、可能な接続構成の中から実際の接続構成をGUI等によりユーザに選択させる処理を追加することも可能である。

【0204】

図119は、図面表示プログラム1104がネットワーク構成図面の描画時にノンインテリジェントハブを予測する処理を示すフローチャートである。

図面表示プログラム1104はノンインテリジェントハブの予測処理要求を待ち(ステップ11901)、ノンインテリジェントハブの予測処理要求としてTSテーブル1125のParent IP Address項目の値(Parent変数に設定)とParent Port項目の値(Pport変数に設定)を受信すると(ステップ11902)、Parent変数とPport変数をキーに指定してTSテーブル1125のParent IP Address項目とParent Port項目を検索し、ヒットするエントリがあるかチェックする(ステップ11903)。

ヒットするエントリがある場合は、Count変数の値をインクリメント(Count変数の初期値は“0”)し(ステップ11904)、ステップ11903から処理を繰り返す。

ヒットするエントリがない場合は、Count変数が“1”より大きいチェックし(ステップ11905)、Count変数が“1”以下の場合はFalseを返し(ステップ11906)、Count変数が“1”より大きい場合はTrueを返す(ステップ11907)。

ステップ11906、ステップ11907の任意のステップ終了後、ステップ

1 1 9 0 1 から処理を繰り返す。

ノンインテリジェントハブ予測処理では1台のネットワーク中継装置の同一ポートに複数の機器が直接接続している場合にノンインテリジェントハブが存在すると予測する。

【 0 2 0 5 】

図 1 2 0 は、図面表示プログラム 1 1 0 4 がユーザイベントにより機器情報を表示する処理を示すフローチャートである。

図面表示プログラム 1 1 0 4 は機器情報表示要求を待ち(ステップ 1 2 0 0 1)、ユーザがマウス等を利用してネットワーク構成図面上の機器の表示部分をクリックするといったユーザのマウスイベントによる機器情報表示要求を受信すると(ステップ 1 2 0 0 2)、ネットワーク構成図面上の機器の表示部分を反転表示するといった G U I 表示を実行した後で、該当するホスト名を取得する(ステップ 1 2 0 0 3)。

取得したホスト名をキーに指定して、T I テーブル 1 1 2 3 の Host Name 項目を検索し、ipaddress 変数にエントリの IP Address 項目の値を設定する(ステップ 1 2 0 0 4)。最後に ipaddress 変数をキーに指定して、A T テーブル 1 1 2 2、T I テーブル 1 1 2 3、T S テーブル 1 1 2 5 を検索し、取得したエントリの情報を機器情報表示部分に描画し(ステップ 1 2 0 0 5)、ステップ 1 2 0 0 1 から処理を繰り返す。

【 0 2 0 6 】

図 1 2 1 は、図面表示プログラム 1 1 0 4 が接続先の変更を監視する処理を示すフローチャートである。

図面表示プログラム 1 1 0 4 は接続先の変更を監視する処理要求を待ち(ステップ 1 2 1 0 1)、接続先の変更を監視する処理要求を受信すると(ステップ 1 2 1 0 2)、図 1 1 6 に示したネットワーク構成図面表示処理を実行し、検出したネットワーク構成を描画する(ステップ 1 2 1 0 3)。

次にネットワーク構成を検出する際に作成した T S テーブルのデータを TS_NEW の領域に格納する(ステップ 1 2 1 0 4)。

ここで、TS_NEW と TS_OLD (初期値は NULL 値)を比較し、エントリ数が減少してい

るかチェックする(ステップ12105)。ただしTS_OLDがNULL値に等しい場合は比較できないので無視する。TS_NEWはTS_OLDよりもエントリ数が減少した場合、ユーザに機器が停止、もしくは接続から外されたことを通知し(ステップ12106)、ステップ12101から繰り返す。

【0207】

TS_NEWはTS_OLDよりもエントリ数が減少していない場合は、TS_NEWとTS_OLDでエントリが置き変わったかチェックし(Terminal IP Address項目の値が等しいがParent IP AddressやParent Portが異なる場合)(ステップ12107)、TS_NEWとTS_OLDでエントリが置き変わった場合はユーザに機器が移動したことを通知し(ステップ12108)、ステップ12101から繰り返す。

TS_NEWとTS_OLDでエントリが置き変わっていない場合は、TS_NEWはTS_OLDよりもエントリ数が増加しているかチェックする(ステップ12109)。TS_NEWはTS_OLDよりもエントリ数が増加している場合はユーザに機器が新規に追加されたことを通知し(ステップ12110)、ステップ12101から繰り返す。

TS_NEWはTS_OLDよりもエントリ数が増加していない場合もステップ12101から繰り返す。

【0208】

以上で説明した本発明の実施形態においては、SNMPマネージャを実装した管理者端末からネットワークノード内の各ネットワーク機器に対してICMPエコーリクエストを送信し、その応答によって稼動状態のネットワーク機器を検出し、その検出した各ネットワーク機器のSNMPエージェントに対し、当該ネットワーク機器内の管理情報ベースの格納情報の転送要求を送信し、返信された管理情報ベースの格納情報によってネットワークノード内に存在するネットワーク機器の種別を検出するようにしているため、SNMP以外の特別なソフトウェアの実装を必要とせず、またSNMPの実装の仕方に依存せずに、少なくとも1台の管理端末からネットワークノード内部の物理的な機器の種別を含む構成を自動的に検出することができる。

【0209】

また、機器種別がブリッジ機能を有するネットワーク機器の管理情報ベースか

ら当該ネットワーク機器の各ポートに接続されたネットワーク機器の物理アドレスの集合を取得すると共に、ルーティング機能を有するネットワーク機器の管理情報ベースから物理アドレスとIPアドレスの対応情報を取得し、それらの取得した物理アドレスとIPアドレスの対応情報に基づき、ブリッジ機能を有するネットワーク機器の各ポートの接続先の機器をIPレベルで認識するようにしたため、ネットワーク機器の各ポートの接続関係をIPレベルで検出することができる。

【0210】

さらに、ICMPエコーリクエストに対して応答が返信されたネットワーク機器は稼動中、応答が返信されないネットワーク機器は存在しないものと認識し、さらに物理アドレスとIPアドレスの対応情報を参照し、稼動中と認識したネットワーク機器以外の対応情報が存在する場合には当該ネットワーク機器は非稼動中であるものと認識するようにしたため、稼動状態のネットワーク機器のみでなく一時的に停止しているネットワーク機器も検出することができる。

【0211】

さらに、ブリッジ機能またはリピータ機能を有するネットワーク機器の管理情報ベースに当該ネットワーク機器の各ポートの接続先に存在する非稼動中のネットワーク機器の情報が格納されているか否かを調べ、格納されている場合は当該格納情報に基づき非稼動中のネットワーク機器の接続関係を検出するようにしたため、非稼動中のネットワーク機器の管理情報ベースの格納情報が取得できない場合であっても、その接続関係を検出することができる。

【0212】

また、ブリッジ機能を有するネットワーク機器が複数存在するか否かを検出し、複数の存在を検出した場合には、任意のブリッジ機能を有するネットワーク機器を親機器とし、該親機器の特定ポートの接続先に別のブリッジ機能を有するネットワーク機器が存在するか否かをさらに検出し、存在することを検出した場合には、該ネットワーク機器を子機器とし、その子機器の各ポートの接続先の機器構成を検索し、ブリッジ機能を有するネットワーク機器同士の接続関係をポート単位で認識するようにしたため、親子関係にある接続関係を検出することができる。

る。

【 0 2 1 3 】

また、子機器と接続している親機器のポートの接続先に存在するネットワーク機器の物理アドレスの集合と、子機器の全ポートから親機器へ接続しているポートを除くポートの接続先に存在するネットワーク機器の物理アドレスの集合の和集合の差分を求め、親機器と子機器の中間に存在しているネットワーク機器を認識するようにしたため、親子関係や兄弟関係にある接続関係を検出することができる。

【 0 2 1 4 】

さらに、親機器と子機器の中間に複数の機器が存在することを認識した場合に、全機器がルーティング機能、ブリッジ機能、リピータ機能にいずれを保持しているか否かを検出し、いずれも保持していない場合にはノンインテリジェントなネットワーク中継装置が存在するものと予測するため、ノンインテリジェントなネットワーク中継装置の存在を検出することができる。

【 0 2 1 5 】

さらに、接続関係を認識した前記親機器と子機器についてそれぞれの管理情報ベース内に保持されている物理アドレスを調べ、親機器の管理情報ベース内に子機器の物理アドレスが保持されていない場合および子機器の管理情報ベース内に親機器の物理アドレスが存在しない場合は、親機器と子機器の特定のポートの接続先に存在する機器の物理アドレスの集合に共通で含まれるような任意の機器を選択し、その選択した機器に対する親機器や子機器の接続ポートを基に親機器と子機器の接続関係を絞り込んで認識するため、管理情報ベースのキャッシュ漏れ等の格納情報の不備に対しても対応することができる。

【 0 2 1 6 】

また、リピータ機能を有するネットワーク機器の任意のポートにおける最新受信フレームの送信元物理アドレスの更新頻度の値を取得し、この値によって当該任意のポートの接続先に稼動している機器の数を認識し、さらに前記更新頻度の値が「0」または「1」以外の場合には当該任意のポートにおける最新受信フレームの送信元物理アドレスの値を所定時間間隔で取得し、当該任意のポートの接

続先に存在する全てのネットワーク機器の物理アドレスを認識するようにしたため、リピータ機能を有するネットワーク機器の任意のポートに接続されているネットワーク機器の数とその物理アドレスを検出することができる。

【 0 2 1 7 】

また、リピータ機能を有するネットワーク機器の任意のポートにおける最新受信フレームの送信元物理アドレスの更新頻度の値を所定時間間隔で取得し、この値が変化しているか否かによってリピータ機能を有するネットワーク機器が R F C 仕様に準拠しているものか否かを検出することができる。

【 0 2 1 8 】

また、ブリッジ機能を有するネットワーク機器およびリピータ機能を有するネットワーク機器の管理情報ベースの格納情報によって接続関係を認識できないネットワーク機器について、前記ブリッジ機能を有するネットワーク機器およびリピータ機能を有するネットワーク機器の任意のポートを一時的に閉塞し、閉塞前には I C M P エコーリクエスト packets に対する応答があり、閉塞後には応答がなくなる機器である場合、その機器は当該任意のポートの接続先に存在するものとして認識することができる。

【 0 2 1 9 】

さらに、ブリッジ機能を有するネットワーク機器およびリピータ機能を有するネットワーク機器の管理情報ベースの格納情報によって接続関係を認識できないネットワーク機器について、前記ブリッジ機能を有するネットワーク機器およびリピータ機能を有するネットワーク機器のポート単位の送受信フレームの統計量を所定時間間隔で収集し、有意の差がないポートの組があれば、このポートの組を接続関係にあるポートとして認識することができる。

【 0 2 2 0 】

また、稼働中のネットワーク機器の管理情報ベースの格納情報を所定時間間隔で収集し、管理者端末の記憶領域に保持し、前回の収集内容と今回の収集内容との相違があるか否かを比較し、稼働中のネットワーク機器の起動、停止、接続先の変更、 I P アドレスの変更等を検出することができる。

【 0 2 2 1 】

さらに、ネットワーク機器同士の接続関係の情報により機器同士の接続関係のモデルを作成し、機器同士の接続関係のモデルごとに、または複数の機器同士の接続関係のモデルを組合せることによってネットワーク機器同士の接続関係を検出したり、検出条件を提示することができる。

【 0 2 2 2 】

なお、上記実施形態は、現在における SNMP プロトコルに則って構成したものであるが、SNMP プロトコルの更新に際しては、細部の構成を変更して実施できることは言うまでもない。

また、ネットワーク機器は有線回線で接続したものに限定されず、無線回線で接続されたものであってもよい。

【 0 2 2 3 】

【発明の効果】

以上の説明から明らかなように、本発明によれば、SNMP を実装しているインテリジェントなネットワーク中継装置が稼動しているネットワーク環境において、SNMP 以外の特別なソフトウェアの実装を必要とせず、また SNMP の実装の仕方に依存せずに、少なくとも 1 台の管理端末からネットワークノード内部の物理的な機器構成を自動的に検出することができる。

【 0 2 2 4 】

また、ブリッジとブリッジの接続先の機器との間に存在するネットワーク機器の検出に限定されることなく、ネットワーク内の全ての機器の種別、接続関係等の構成を検出することができる。

【 0 2 2 5 】

さらに、ハブ同士がカスケード接続されている場合やリピータの接続先に複数の端末が接続されている場合であっても、その接続関係を検出することができる。

【 0 2 2 6 】

また、SNMP プロトコルを実装していないノンインテリジェント機器であっても、その存在を予測することができるなどの効果が得られる。

【図面の簡単な説明】

【図 1】

本発明に係わるネットワーク構成自動認識方法で対象とするネットワークシステムの実施形態を示す図である。

【図 2】

本発明に係わるネットワーク構成自動認識方法で用いる SNMP メッセージフォーマットを示す図である。

【図 3】

本発明に係わるネットワーク構成自動認識方法で用いるインターネット O I D ツリーを示す図である。

【図 4】

本発明に係わるネットワーク構成自動認識方法で用いる M I B オブジェクトの構成を示す図である。

【図 5】

本発明に係わるネットワーク構成自動認識方法で用いる system グループオブジェクトの構成を示す図である。

【図 6】

本発明に係わるネットワーク構成自動認識方法で用いる interfaces グループオブジェクトの構成を示す図である。

【図 7】

本発明に係わるネットワーク構成自動認識方法で用いる ip グループオブジェクトの構成を示す図である。

【図 8】

本発明に係わるネットワーク構成自動認識方法で用いる dot1dBridge グループオブジェクトの構成を示す図である。

【図 9】

本発明に係わるネットワーク構成自動認識方法で用いる snmpDot3RptrMgt グループオブジェクトの構成を示す図である。

【図 1 0】

本発明に係わるネットワーク構成自動認識方法で用いる printmib グループオブ

ジェクトの構成を示す図である。

【図 1 1】

本発明に係わるネットワーク構成自動認識方法を実現する管理者端末内のプログラム構成例を示す図である。

【図 1 2】

本発明に係わるネットワーク構成自動認識方法で用いる O I D テーブルの構成を示す図である。

【図 1 3】

本発明に係わるネットワーク構成自動認識方法で用いる A T テーブルの構成を示す図である。

【図 1 4】

本発明に係わるネットワーク構成自動認識方法で用いる T I テーブル 1 1 2 3 の構成を示す図である。

【図 1 5】

本発明に係わるネットワーク構成自動認識方法で用いる P F テーブル 1 1 2 4 の構成を示す図である。

【図 1 6】

本発明に係わるネットワーク構成自動認識方法で用いる T S テーブルの構成を示す図である。

【図 1 7】

本発明に係わるネットワーク構成自動認識方法における S N M P メッセージ送受信の仕組みを示す図である。

【図 1 8】

本発明に係わるネットワーク構成自動認識方法における機器種別の検出方法を説明する図である。

【図 1 9】

本発明に係わるネットワーク構成自動認識方法を考慮する場合のネットワーク中継装置間の Relation 定義を示す図である。

【図 2 0】

本発明に係わるネットワーク構成自動認識方法における interfaces M I B を利用したネットワーク中継装置間の接続検出方法を示す図である。

【図 2 1】

本発明に係わるネットワーク構成自動認識方法におけるネットワーク機器の分類方法を示す図である。

【図 2 2】

本発明に係わるネットワーク構成自動認識方法における R-CF-CF モデルの接続検出の仕組を示す図である。

【図 2 3】

本発明に係わるネットワーク構成自動認識方法における R-CF-CF モデルの接続検出に利用する P F テーブルのエントリ例を示す図である。

【図 2 4】

本発明に係わるネットワーク構成自動認識方法における R-CF-IF モデルの接続検出の仕組を示す図である。

【図 2 5】

本発明に係わるネットワーク構成自動認識方法における R-CF-IF モデルの接続検出に利用する P F テーブルのエントリ例を示す図である。

【図 2 6】

本発明に係わるネットワーク構成自動認識方法における R-CF-SF モデルの接続検出の仕組を示す図である。

【図 2 7】

本発明に係わるネットワーク構成自動認識方法における R-CF-SF モデルの接続検出に利用する P F テーブルのエントリ例を示す図である。

【図 2 8】

本発明に係わるネットワーク構成自動認識方法における R-IF-CF モデルの接続検出の仕組を示す図である。

【図 2 9】

本発明に係わるネットワーク構成自動認識方法における R-IF-CF モデルの接続検出に利用する P F テーブルのエントリ例を示す図である。

【図 3 0】

本発明に係わるネットワーク構成自動認識方法におけるR-IF-IFモデルの接続検出の仕組みを示す図である。

【図 3 1】

本発明に係わるネットワーク構成自動認識方法におけるR-IF-IFモデルの接続検出に利用するP Fテーブルのエントリ例を示す図である。

【図 3 2】

本発明に係わるネットワーク構成自動認識方法におけるR-IF-SFモデルの接続検出の仕組みを示す図である。

【図 3 3】

本発明に係わるネットワーク構成自動認識方法におけるR-IF-SFモデルの接続検出に利用するP Fテーブルのエントリ例を示す図である。

【図 3 4】

本発明に係わるネットワーク構成自動認識方法におけるR-SF-CFモデルの接続検出の仕組みを示す図である。

【図 3 5】

本発明に係わるネットワーク構成自動認識方法におけるR-SF-CFモデルの接続検出に利用するP Fテーブルのエントリ例を示す図である。

【図 3 6】

本発明に係わるネットワーク構成自動認識方法におけるR-SF-IFモデルの接続検出の仕組みを示す図である。

【図 3 7】

本発明に係わるネットワーク構成自動認識方法におけるR-SF-IFモデルの接続検出に利用するP Fテーブルのエントリ例を示す図である。

【図 3 8】

本発明に係わるネットワーク構成自動認識方法におけるR-SF-SFモデルの接続検出の仕組みを示す図である。

【図 3 9】

本発明に係わるネットワーク構成自動認識方法におけるR-SF-SFモデルの接続

検出に利用する P F テーブルのエントリ例を示す図である。

【図 4 0】

本発明に係わるネットワーク構成自動認識方法における R-CF モデルの接続検出の仕組を示す図である。

【図 4 1】

本発明に係わるネットワーク構成自動認識方法における R-CF モデルの接続検出に利用する P F テーブルのエントリ例を示す図である。

【図 4 2】

本発明に係わるネットワーク構成自動認識方法における R-IF モデルの接続検出の仕組を示す図である。

【図 4 3】

本発明に係わるネットワーク構成自動認識方法における R-IF モデルの接続検出に利用する P F テーブルのエントリ例を示す図である。

【図 4 4】

本発明に係わるネットワーク構成自動認識方法における R-SF モデルの接続検出の仕組を示す図である。

【図 4 5】

本発明に係わるネットワーク構成自動認識方法における R-SF モデルの接続検出に利用する P F テーブルのエントリ例を示す図である。

【図 4 6】

本発明に係わるネットワーク構成自動認識方法におけるネットワーク中継装置同士の接続検出方法を説明する図である。

【図 4 7】

図 4 6 の続きを示す図である。

【図 4 8】

本発明に係わるネットワーク構成自動認識方法における CF-Term モデルの接続検出の仕組を示す図である。

【図 4 9】

本発明に係わるネットワーク構成自動認識方法における CF-Term モデルの接続

検出に利用する P F テーブルのエントリ例を示す図である。

【図 5 0】

本発明に係わるネットワーク構成自動認識方法における IF-Term モデルの接続検出の仕組みを示す図である。

【図 5 1】

本発明に係わるネットワーク構成自動認識方法における IF-Term モデルの接続検出に利用する P F テーブルのエントリ例を示す図である。

【図 5 2】

本発明に係わるネットワーク構成自動認識方法における SF-Term モデルの接続検出の仕組みを示す図である。

【図 5 3】

本発明に係わるネットワーク構成自動認識方法における SF-Term モデルの接続検出に利用する P F テーブルのエントリ例を示す図である。

【図 5 4】

本発明に係わるネットワーク構成自動認識方法におけるネットワーク中継装置と端末装置の接続検出方法を説明する図である。

【図 5 5】

本発明に係わるネットワーク構成自動認識方法における複数のモデルを組み合わせることによる親子関係の検出方法を説明する図である。

【図 5 6】

本発明に係わるネットワーク構成自動認識方法における複数のモデルを組み合わせることによる親子関係の検出に利用する T S テーブルのエントリ例を示す図である。

【図 5 7】

本発明に係わるネットワーク構成自動認識方法における Non Intelligent Hub の接続の予測方法を示す図である。

【図 5 8】

本発明に係わるネットワーク構成自動認識方法における Non Intelligent Hub の接続の予測に利用する T S テーブルのエントリ例を示す図である。

【図 5 9】

本発明に係わるネットワーク構成自動認識方法における非稼動中端末装置の検出方法を示す図である。

【図 6 0】

本発明に係わるネットワーク構成自動認識方法における接続先の変更の検出方法を示す図である。

【図 6 1】

本発明に係わるネットワーク構成自動認識方法における接続先の変更の検出に利用する T S テーブルのエントリ例を示す図である。

【図 6 2】

本発明に係わるネットワーク構成自動認識方法におけるネットワーク構成図面表示例を示す図である。

【図 6 3】

本発明に係わるネットワーク構成自動認識方法における稼動状況検出モジュールが I C M P エコーリクエストを送受信する処理を示すフローチャートである。

【図 6 4】

本発明に係わるネットワーク構成自動認識方法における M I B アクセスモジュールが P D U を作成し、 S N M P メッセージを送受信する処理を示すフローチャートである。

【図 6 5】

本発明に係わるネットワーク構成自動認識方法における M I B アクセスモジュールが機器の M I B 2 サポート状況をチェックする処理 I P フォワーディング機能の有無をチェックする処理を示すフローチャートである。

【図 6 6】

本発明に係わるネットワーク構成自動認識方法における M I B アクセスモジュールが機器のブリッジ M I B サポート状況をチェックする処理を示すフローチャートである。

【図 6 7】

本発明に係わるネットワーク構成自動認識方法における M I B アクセスモジュ

ールが機器のリピータM I Bサポート状況をチェックする処理を示すフローチャートである。

【図 6 8】

本発明に係わるネットワーク構成自動認識方法におけるM I Bアクセスモジュールが機器のプリンタM I Bサポート状況をチェックする処理を示すフローチャートである。

【図 6 9】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがA Tテーブルを作成する処理を示すフローチャートである。

【図 7 0】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがT Iテーブルを作成する処理を示すフローチャートである。

【図 7 1】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがT Iテーブル作成時にT Iテーブルの各項目の値を取得する処理を示すフローチャートである。

【図 7 2】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがT Iテーブル作成時に機器タイプを認識する処理を示すフローチャートである。

【図 7 3】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがP Fテーブルを作成する処理を示すフローチャートである。

【図 7 4】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがP Fテーブル作成時にブリッジM I Bサポート機器に対して実行する処理を示すフローチャートである。

【図 7 5】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモ

ジュールが P F テーブル作成時にリピータ M I B サポート機器に対して実行する処理を示すフローチャートである。

【図 7 6】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールが P F テーブル作成時にフォワーディング情報を学習する処理を示すフローチャートである。

【図 7 7】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールが P F テーブル作成時にフォワーディング情報を予測する処理を示すフローチャートである。

【図 7 8】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールが P F テーブル作成時に M I B 2 (interfaces M I B) サポート機器に対して実行する処理を示すフローチャートである。

【図 7 9】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールが P F テーブル作成時に管理者端末の接続ポートを検出する処理を示すフローチャートである。

【図 8 0】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールが P F テーブル作成時に管理者端末以外の機器の接続ポートを検出する処理を示すフローチャートである。

【図 8 1】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールが T S テーブルを作成する処理を示すフローチャートである。

【図 8 2】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールが T S テーブル作成時に Root 装置を決定する処理を示すフローチャートである。

【図 8 3】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがTSテーブル作成時にネットワーク中継装置間の接続を決定する処理を示すフローチャートである。

【図 8 4】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがTSテーブル作成時に接続モデルを決定する処理を示すフローチャートである。

【図 8 5】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがTSテーブル作成時にネットワーク機器を分類する処理を示すフローチャートである。

【図 8 6】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがTSテーブル作成時に接続検出条件をチェックする処理を示すフローチャートである。

【図 8 7】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがTSテーブル作成時に集合(R, CF, CF)の接続検出条件をチェックする処理を示すフローチャートである。

【図 8 8】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがTSテーブル作成時に集合(R, CF, IF)の接続検出条件をチェックする処理を示すフローチャートである。

【図 8 9】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがTSテーブル作成時にR-IF-CFモデルの接続検出条件をチェックする処理を示すフローチャートである。

【図 9 0】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがTSテーブル作成時にR-CF-IFモデルの接続検出条件をチェックする処理を示すフローチャートである。

【図 9 1】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがTSテーブル作成時に集合(R, CF, SF)の接続検出条件をチェックする処理を示すフローチャートである。

【図 9 2】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがTSテーブル作成時にR-SF-CFモデルの接続検出条件をチェックする処理を示すフローチャートである。

【図 9 3】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがTSテーブル作成時にR-CF-SFモデルの接続検出条件をチェックする処理を示すフローチャートである。

【図 9 4】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがTSテーブル作成時に集合(R, IF, IF)の接続検出条件をチェックする処理を示すフローチャートである。

【図 9 5】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがTSテーブル作成時にR-IF-IFモデルの接続検出条件をチェックする処理を示すフローチャートである。

【図 9 6】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがTSテーブル作成時に集合(R, IF, SF)の接続検出条件をチェックする処理を示すフローチャートである。

【図 9 7】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモ

ジュールがTSテーブル作成時にR-SF-IFモデルの接続検出条件をチェックする処理を示すフローチャートである。

【図 9 8】

図 9 7 の続きを示すフローチャートである。

【図 9 9】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがTSテーブル作成時にR-IF-SFモデルの接続検出条件をチェックする処理を示すフローチャートである。

【図 1 0 0】

図 9 9 の続きを示すフローチャートである。

【図 1 0 1】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがTSテーブル作成時に集合(R, SF, SF)の接続検出条件をチェックする処理を示すフローチャートである。

【図 1 0 2】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがTSテーブル作成時にTSテーブルに対するエントリを追加する処理を示すフローチャートである。

【図 1 0 3】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがTSテーブル作成時にTSテーブルに対するRootエントリを追加する処理を示すフローチャートである。

【図 1 0 4】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがTSテーブル作成時に親子関係と接続関係が不明なネットワーク中継装置を追加する処理を示すフローチャートである。

【図 1 0 5】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがTSテーブル作成時に親子関係だけが不明なネットワーク中継装置を

追加する処理を示すフローチャートである。

【図 1 0 6】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがTSテーブル作成時に親子関係と接続関係が自明なネットワーク中継装置を追加する処理を示すフローチャートである。

【図 1 0 7】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがTSテーブル作成時に親子関係を決定する処理を示すフローチャートである。

【図 1 0 8】

図 1 0 7 の続きを示すフローチャートである。

【図 1 0 9】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがTSテーブル作成時に複数モデルを組合わせる処理を示すフローチャートである。

【図 1 1 0】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがTSテーブル作成時にTSテーブルのエントリを結合する処理を示すフローチャートである。

【図 1 1 1】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがTSテーブル作成時に接続関係が不明なネットワーク中継装置を決定する処理を示すフローチャートである。

【図 1 1 2】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがTSテーブル作成時にRootとネットワーク中継装置の親子関係を決定する処理を示すフローチャートである。

【図 1 1 3】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモ

ジュールがT Sテーブル作成時にRootとネットワーク中継装置の接続ポートを決定する処理を示すフローチャートである。

【図 1 1 4】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがT Sテーブル作成時にネットワーク中継装置と端末装置の接続を決定する処理を示すフローチャートである。

【図 1 1 5】

本発明に係わるネットワーク構成自動認識方法におけるオートディスカバリモジュールがT Sテーブル作成時にインタフェースM I Bを評価する処理を示すフローチャートである。

【図 1 1 6】

本発明に係わるネットワーク構成自動認識方法における図面表示プログラムがネットワーク構成図面を表示する処理を示すフローチャートである。

【図 1 1 7】

本発明に係わるネットワーク構成自動認識方法における図面表示プログラムがネットワーク構成図面表示処理時に画面に描画する処理を示すフローチャートである。

【図 1 1 8】

図 1 1 7 の続きを示すフローチャートである。

【図 1 1 9】

本発明に係わるネットワーク構成自動認識方法における図面表示プログラムがネットワーク構成図面の描画時にノンインテリジェントハブを予測する処理を示すフローチャートである。

【図 1 2 0】

本発明に係わるネットワーク構成自動認識方法における図面表示プログラムがユーザイベントにより機器情報を表示する処理を示すフローチャートである。

【図 1 2 1】

本発明に係わるネットワーク構成自動認識方法における図面表示プログラムが接続先変更の監視する処理を示すフローチャートである。

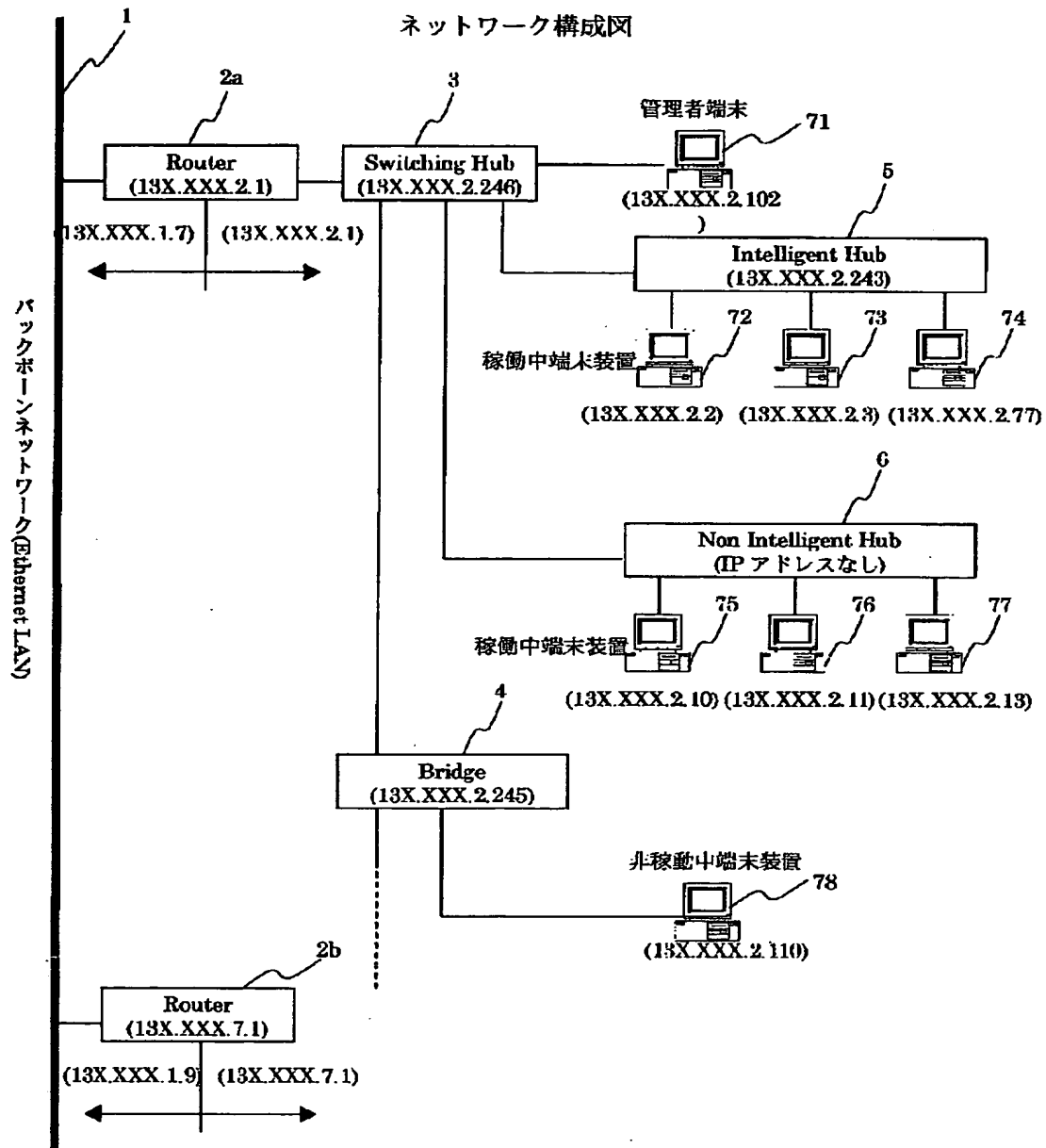
【符号の説明】

1 …バックボーンネットワークケーブル(Ethernet LAN)、2 a, 2 b …ルータ、3 …スイッチングハブ、4 …ブリッジ、5 …インテリジェントハブ、6 …ノンインテリジェントハブ、7 1 …管理者端末、7 2 ～7 8 …端末装置、1 1 0 2 …通信ポート、1 1 0 3 …ネットワーク構成自動認識サービスプログラム、1 1 0 4 …図面表示プログラム、1 1 1 1 …稼動状況検出モジュール、1 1 1 2 …M I B アクセスモジュール、1 1 1 3 …オートディスカバリモジュール、1 1 2 1 …O I D テーブル、1 1 2 2 …A T テーブル、1 1 2 3 …T I テーブル、1 1 2 4 …P F テーブル、1 1 2 5 …T S テーブル。

【書類名】 図面

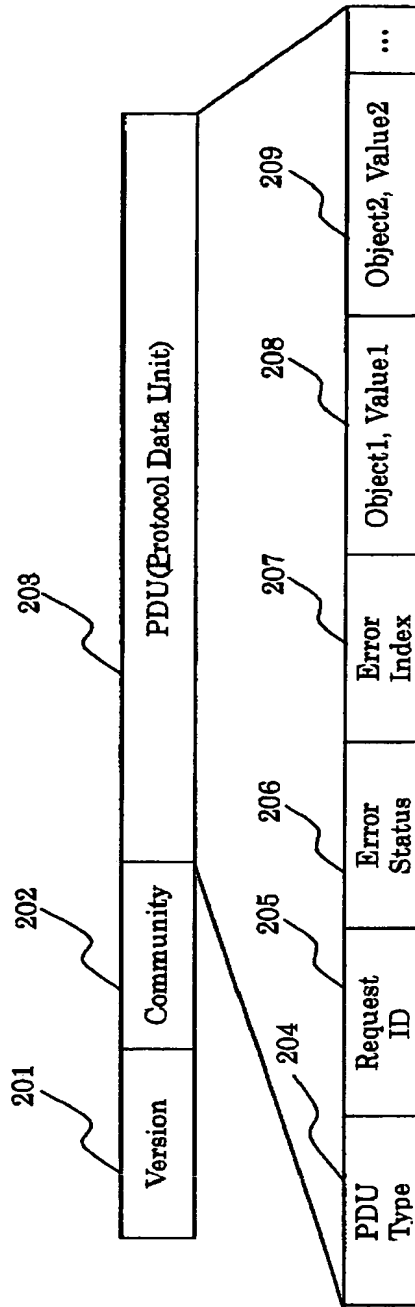
【図 1】

図 1

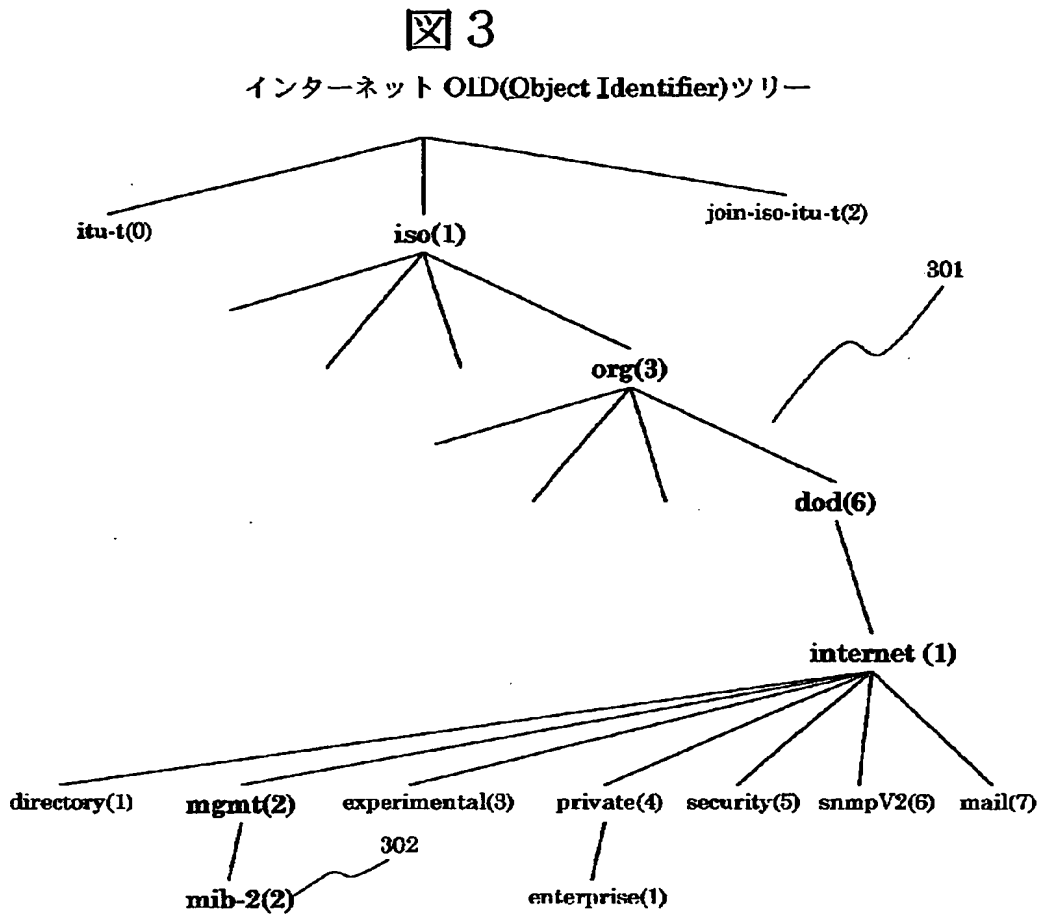


【図 2】

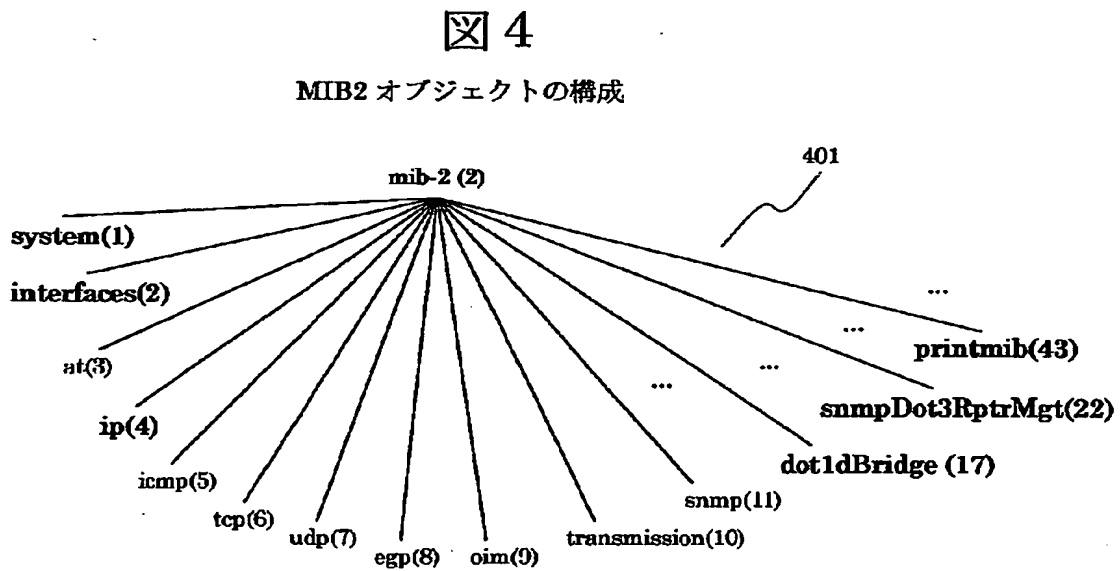
図 2
SNMP メッセージフォーマット



【図 3】



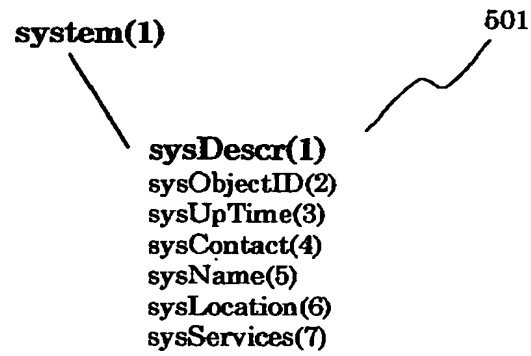
【図 4】



【図 5】

図 5

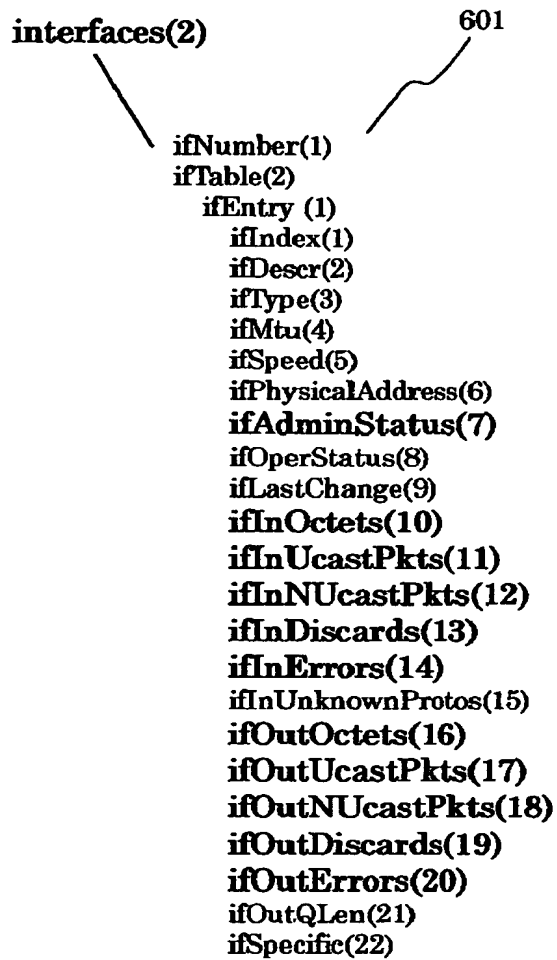
system グループオブジェクトの構成



【図 6】

図 6

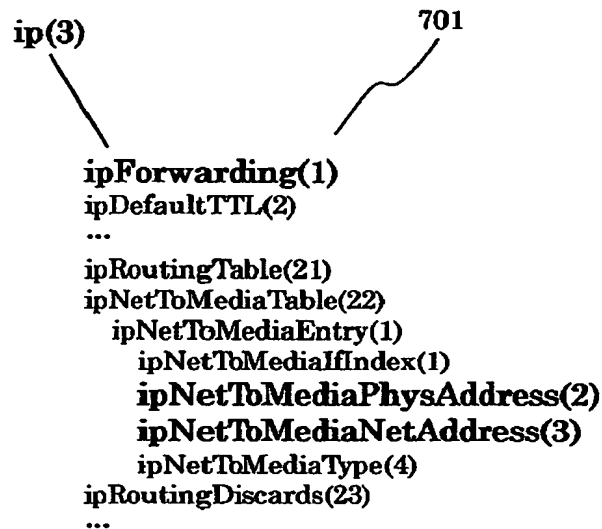
interfaces グループオブジェクトの構成



【図 7】

図 7

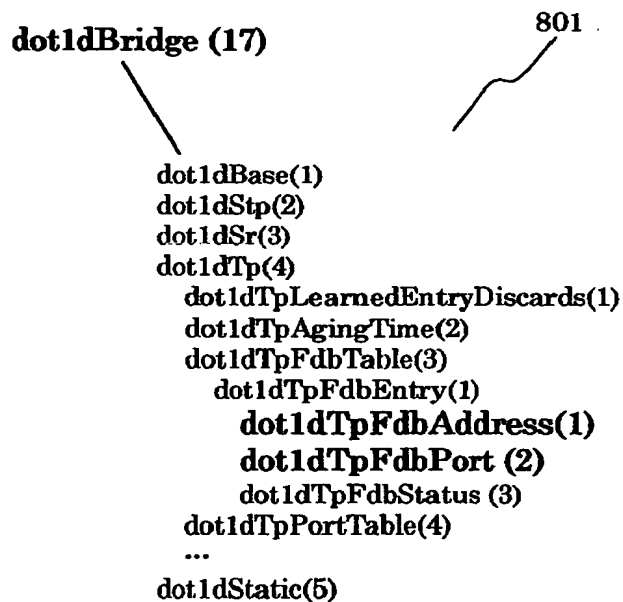
ip グループオブジェクトの構成



【図 8】

図 8

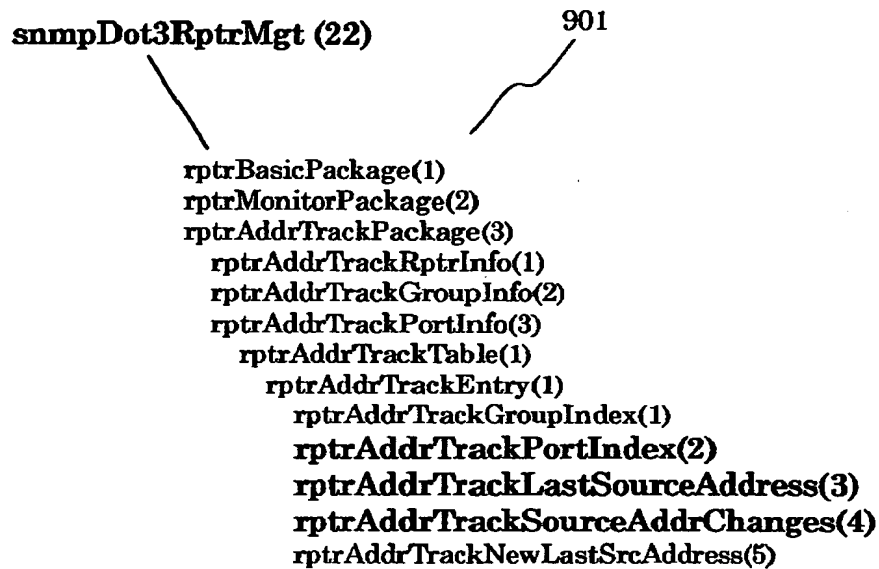
dot1dBridge グループオブジェクトの構成



【図 9】

図 9

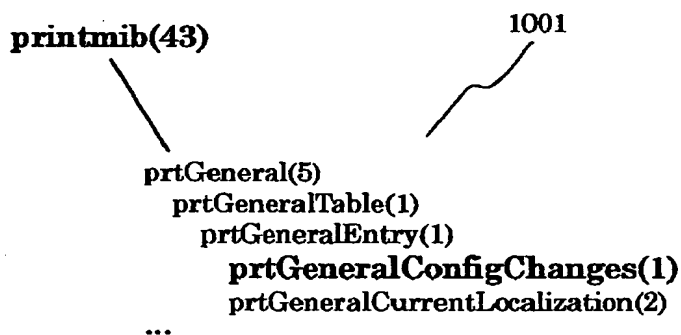
snmpDot3RptrMgt グループオブジェクトの構成



【図 1 0】

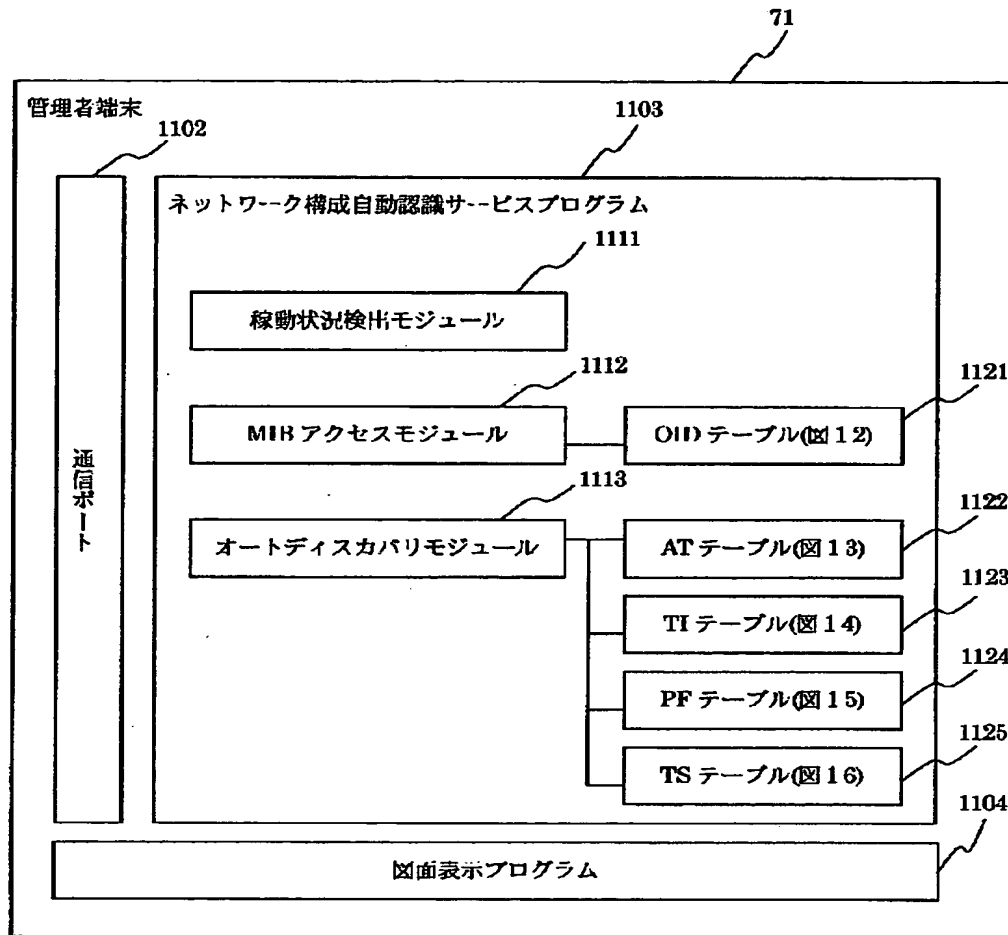
図 1 0

printmib グループオブジェクトの構成



【図 1 1】

図 1 1
プログラム構成図



【図 1 2】

図 1 2

OID(Object Identifier)テーブル構成図

Object Name	Object Identifier	type	Object Path
sysDescr	43.6.1.2.1.1.1.0	String	system.sysDescr
sysObjectID	43.6.1.2.1.1.2.0	Binary	system.sysObjectID
...

【図 1 3】

図 1 3

AT(Address Translation)テーブル構成図

IP Address	Mac Address
13X.XXX.2.1	00:e0:f7:26:a4:e3
13X.XXX.2.2	08:00:20:11:ee:73
...	...

【図 14】

図 14

TI(Terminal Information)テーブル構成図

1123

1401	1402	1403	1404	1405	1406	1407	1408	1409	1410
IP Address	Mac Address	Host Name	type	alive	mib2	forwarding	bridge	repeater	print
13X.XXX.2.	00:e0:f7:26:a4:e3	ori-irouter.ori.xxx.co.jp	R	On	On	On	On	Off	Off
13X.XXX.2.	08:00:20:a1:33:ab	ori.ori.xxx.co.jp	T	On	On	Off	Off	Off	Off
13X.XXX.2.	-	-	-	On	Off	Off	Off	Off	Off
...

(U: Unknown; 0: Router; 1: SH: Switching Hub; 2: IH: Intelligent Hub; 3:
B: Bridge; 4: R: Repeater; 5: T: Terminal; 6: P: Printer; 7) (On: 1, Off: 0)

【図 15】

図 15

PF(Port Forwarding)テーブル構成図

1124

1501	1502	1503	1504	1505
Source IP Address	Source Mac Address	Source Port	Destination IP Address	Destination Mac Address
...	...	2	13X.XXX.2.2	08:00:20:a1:33:ab
13X.XXX.2.1	00:e0:f7:26:a4:e3	2	13X.XXX.2.102	00:e0:18:00:27:d7
...
...	...	2	13X.XXX.2.246	08:00:4e:4f:ad:27
...
13X.XXX.2.248	08:00:4e:4f:ad:27	2	13X.XXX.2.1	00:e0:f7:26:a4:e3
...

【図 16】

図 16

TS(Tree Structure)テーブル構成図

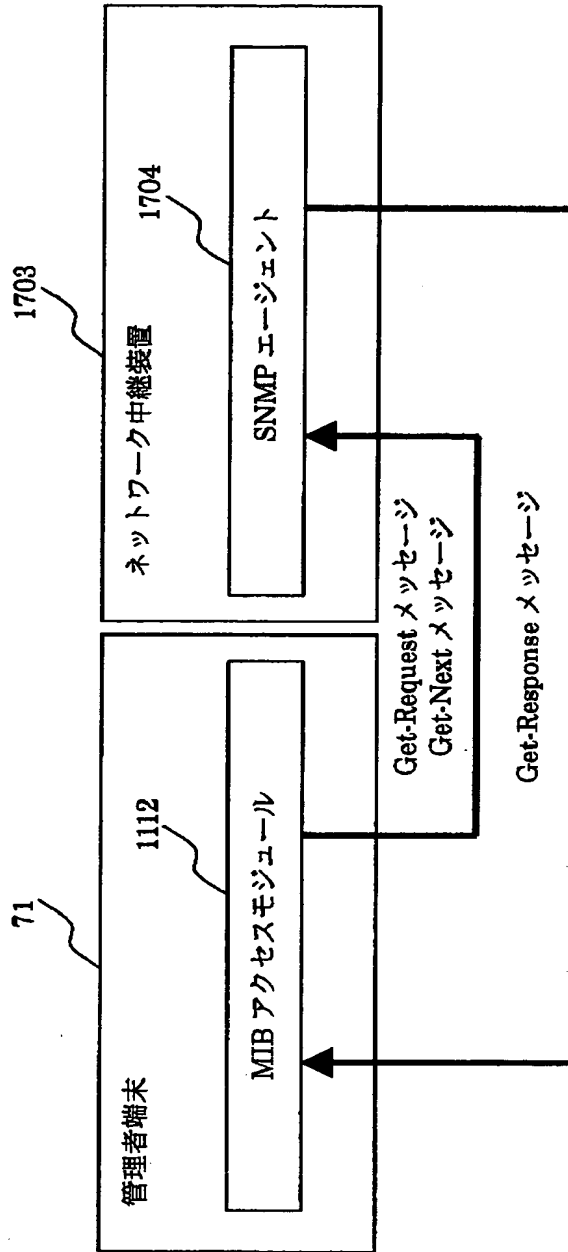
1125

1601	1602	1603	1604	1605	1606
Terminal IP Address	Terminal Mac Address	Terminal Port	Parent IP Address	Parent Mac Address	Parent Port
13X.XXX.2.1	00:e0:f7:26:a4:e3	-	-	-	-
13X.XXX.2.246	08:00:4e:4f:ad:27	2	13X.XXX.2.1	00:e0:f7:26:a4:e3	2
13X.XXX.2.102	00:e0:18:00:27:d7	-	13X.XXX.2.246	08:00:4e:4f:ad:27	3
...
13X.XXX.2.243	00:00:f4:71:01:37	1	13X.XXX.2.246	08:00:4e:4f:ad:27	1
...
13X.XXX.2.2	08:00:20:a1:33:ab	-	13X.XXX.2.243	00:00:f4:71:01:37	2
...

【図17】

図 17

SNMP メッセージ送受信の仕組



【図 18】

図 18

機器種別の検出方法

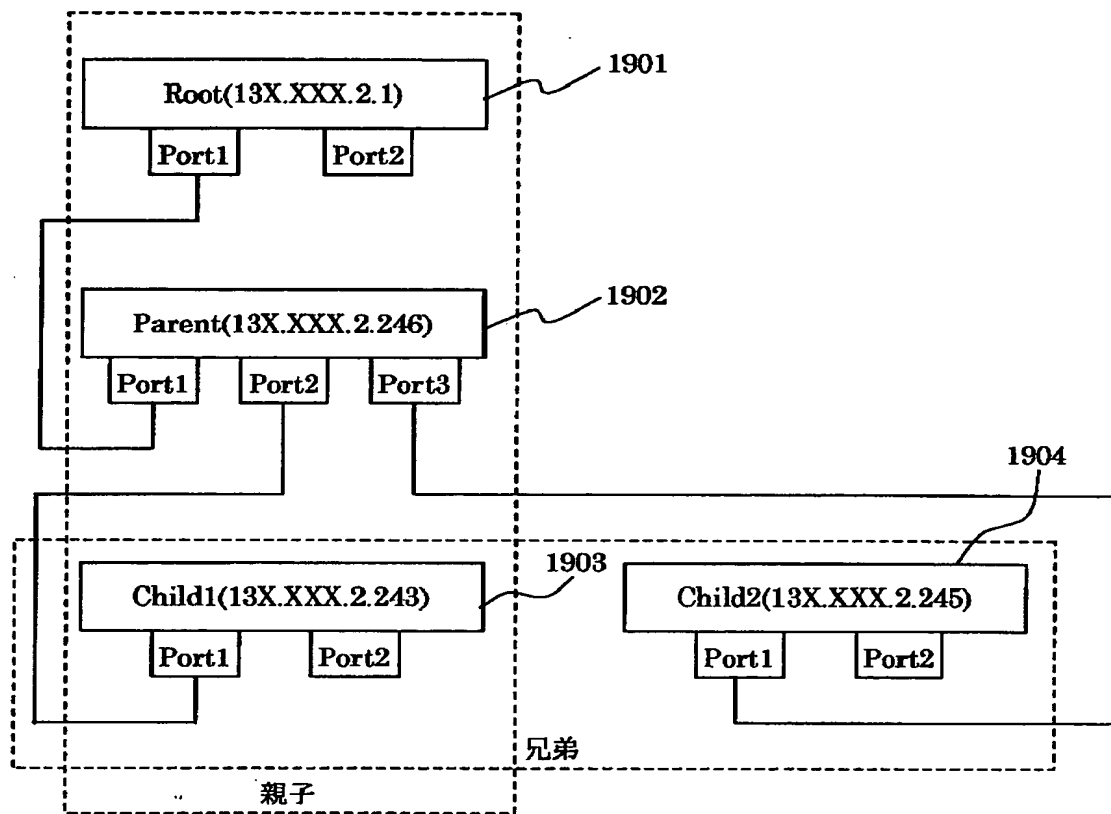
機器 MIB	1801	1802	1803	1804	1805	1806	1807
	Router	Bridge	Switching Hub	Intelligent Hub	Non Intelligent Hub (Repeater)	Printer	Terminal
ip グループ ipForwarding オブジェクト	○(値=1)	○(値=0)	○(値=1 or 値=0)	○(値=0)	—	○(値=0)	○(値=0)
dot1dBridge グループ 任意オブジェクト	○	○	○	×	—	×	×
snmpDot3RptrMgt グループ 任意オブジェクト	×	×	○	○	—	×	×
printmib グループ 任意オブジェクト	×	×	×	×	—	○	×

注)○:実装、×:非実装、—:MIB 非サポート)

【図 19】

図 19

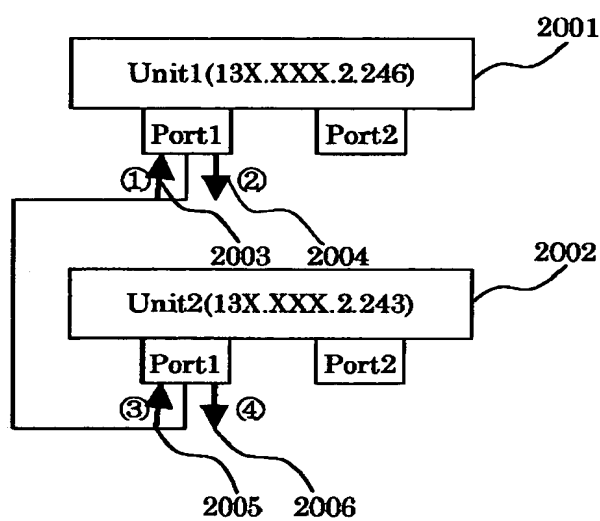
ネットワーク中継装置 Relation 定義図



【図 2 0】

図 2 0

interfaces MIB を利用したネットワーク中継装置間の接続検出



[取得情報]

- ① ifInOctets(Unit1 の Port1)
- ② ifOutOctets(Unit1 の Port1)
- ③ ifInOctets(Unit2 の Port1)
- ④ ifOutOctets(Unit2 の Port1)

[検出条件]

- ・①と④に有意な差がない
 - ・②と③に有意な差がない
- Unit1 の Port1 と Unit2 の Port2
に接続関係がある。

【図 2 1】

図 2 1

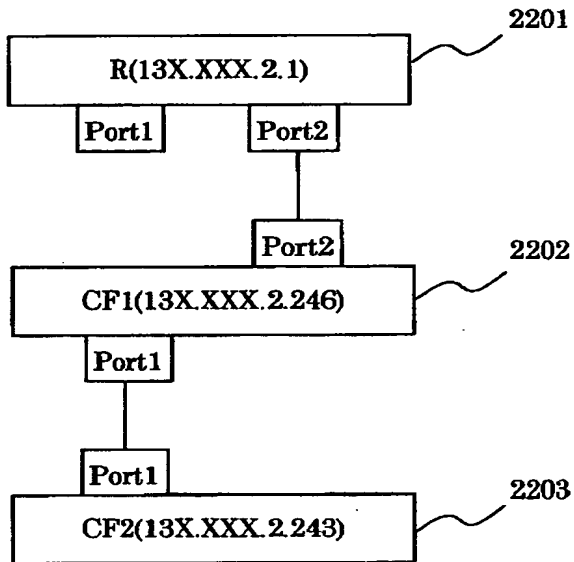
ネットワーク機器の分類方法

ネットワーク 中継装置	意味	
R	セグメントを分割しているネットワーク中継装置(Router)	2101
CF	MIB のオブジェクトの格納情報に不備がなく、 すべてのネットワーク中継装置と端末装置の接続ポートを 格納した PF テーブルを作成可能なネットワーク中継装置	2102
IF	MIB のオブジェクトの格納情報に不備があり、R を除く 他のネットワーク中継装置への接続ポート番号を検出できない 場合が存在するネットワーク中継装置	2103
SF	MIB のオブジェクトの格納情報に不備があり、他のすべて のネットワーク中継装置への接続ポートが検出できず、1 台 以上の端末装置への接続ポートが検出可能なネットワーク 中継装置	2104
NF	MIB を保持していないネットワーク中継装置 (Non Intelligent Hub、Repeater)	2105
Term	ネットワーク中継装置以外の機器 (Printer、Terminal)	2106

【図 2 2】

図 2 2

R-CF-CF モデルの接続検出の仕組



【図 2 3】

図 2 3

R-CF-CF モデルの接続検出に利用する PF テーブルのエントリ 1124

Source IP Address	Source Mac Address	Source Port	Destination IP Address	Destination Mac Address
...
13X.XXX.2.246	08:00:4e:4f:ad:27	1	13X.XXX.2.243	00:00:f4:71:01:37
13X.XXX.2.246	08:00:4e:4f:ad:27	2	13X.XXX.2.1	00:e0:f7:26:a4:e3
...
13X.XXX.2.243	00:00:f4:71:01:37	1	13X.XXX.2.1	00:e0:f7:26:a4:e3
13X.XXX.2.243	00:00:f4:71:01:37	1	13X.XXX.2.246	08:00:4e:4f:ad:27
...

2301

2302

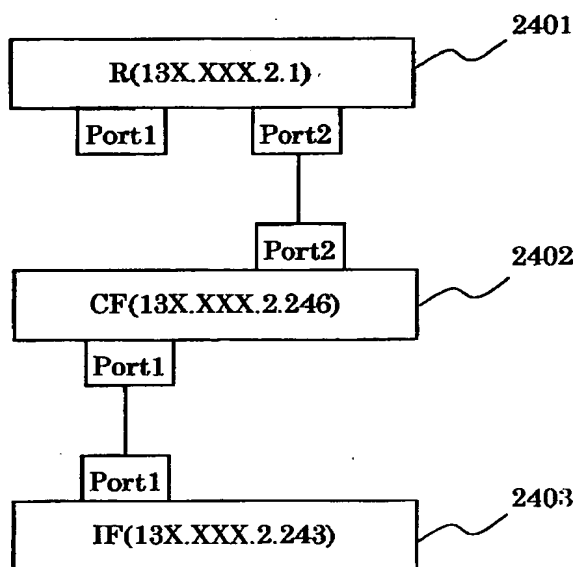
2303

2304

【図 2 4】

図 2 4

R-CF-IF モデルの接続検出の仕組



【図 2 5】

図 2 5

R-CF-IF モデルの接続検出に利用する PF テーブルのエントリ

Source IP Address	Source Mac Address	Source Port	Destination IP Address	Destination Mac Address
...
13X.XXX.2.246	08:00:4e:4f:ad:27	1	13X.XXX.2.243	00:00:f4:71:01:37
13X.XXX.2.246	08:00:4e:4f:ad:27	2	13X.XXX.2.1	00:e0:f7:26:a4:e3
...
13X.XXX.2.243	00:00:f4:71:01:37	1	13X.XXX.2.1	00:e0:f7:26:a4:e3
...

2501

2502

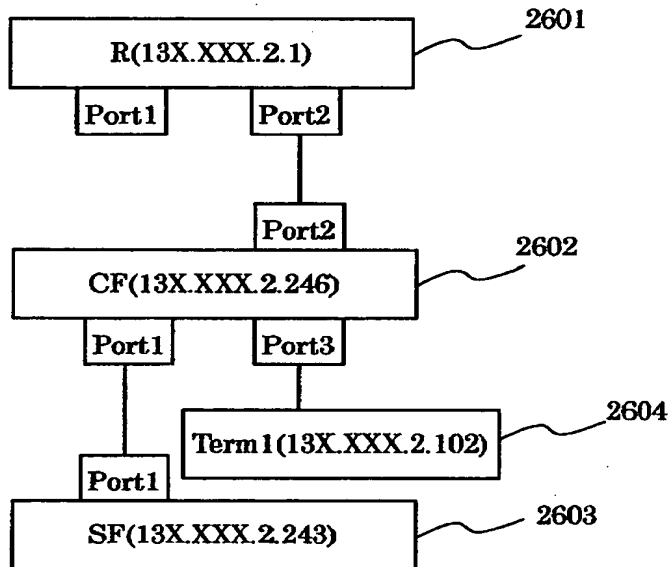
2503

1124

【図 2 6】

図 2 6

R-CF-SF モデルの接続検出の仕組



【図 27】

図 27

R-CF-SF モデルの接続検出に利用する PF テーブルのエントリ

1124

Source IP Address	Source Mac Address	Source Port	Destination IP Address	Destination Mac Address
...
13X.XXX.2.246	08:00:4e:4f:ad:27	1	13X.XXX.2.243	00:00:f4:71:01:37
13X.XXX.2.246	08:00:4e:4f:ad:27	2	13X.XXX.2.1	00:e0:f7:26:a4:e3
13X.XXX.2.246	08:00:4e:4f:ad:27	3	13X.XXX.2.102	00:e0:18:00:27:d7
...
13X.XXX.2.243	00:00:f4:71:01:37	1	13X.XXX.2.102	00:e0:18:00:27:d7
...

2701

2702

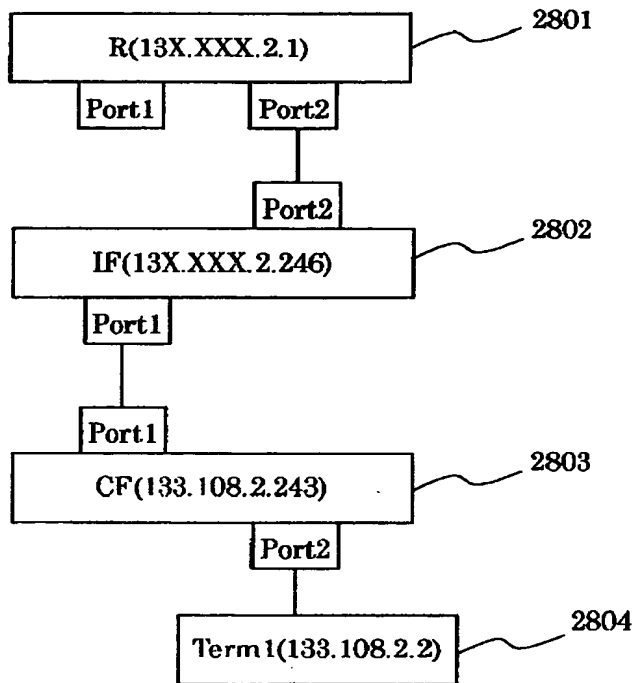
2703

2704

【図 2 8】

図 2 8

R-IF-CF モデルの接続検出の仕組



【図 2 9】

図 2 9

R-IF-CF モデルの接続検出に利用する PF テーブルのエントリ

1124

Source IP Address	Source Mac Address	Source Port	Destination IP Address	Destination Mac Address
...
13X.XXX.2.246	08:00:4e:4f:ad:27	1	13X.XXX.2.2	08:00:20:a1:33:ab
13X.XXX.2.246	08:00:4e:4f:ad:27	2	13X.XXX.2.1	00:e0:f7:26:a4:e3
...
13X.XXX.2.243	00:00:f4:71:01:37	1	13X.XXX.2.1	00:e0:f7:26:a4:e3
13X.XXX.2.243	00:00:f4:71:01:37	1	13X.XXX.2.246	08:00:4e:4f:ad:27
13X.XXX.2.243	00:00:f4:71:01:37	2	13X.XXX.2.2	08:00:20:a1:33:ab
...

2901

2902

2903

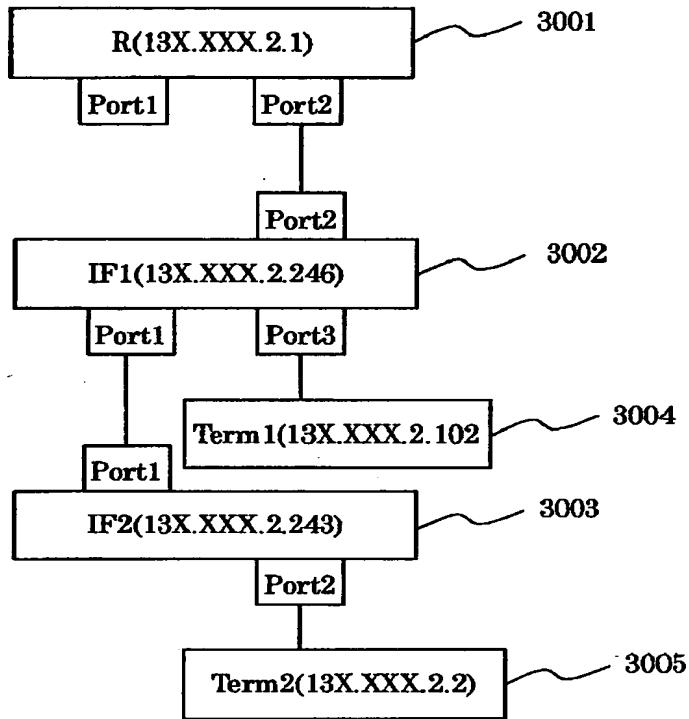
2904

2905

【図 3 0】

図 3 0

R-IF-IF モデルの接続検出の仕組



【図 3 1】

図 3 1

R-IF-IF モデルの接続検出に利用する PF テーブルのエントリ

1124

Source IP Address	Source Mac Address	Source Port	Destination IP Address	Destination Mac Address
...
13X.XXX.2.246	08:00:4e:4f:ad:27	1	13X.XXX.2.2	08:00:20:a1:33:ab
13X.XXX.2.246	08:00:4e:4f:ad:27	2	13X.XXX.2.1	00:e0:f7:26:a4:e3
13X.XXX.2.246	08:00:4e:4f:ad:27	3	13X.XXX.2.102	00:e0:18:00:27:d7
...
13X.XXX.2.243	00:00:f4:71:01:37	1	13X.XXX.2.1	00:e0:f7:26:a4:e3
13X.XXX.2.243	00:00:f4:71:01:37	1	13X.XXX.2.102	00:e0:18:00:27:d7
13X.XXX.2.243	00:00:f4:71:01:37	2	13X.XXX.2.2	08:00:20:a1:33:ab
...

3101

3102

3103

3104

3105

3106

【図 3 2】

図 3 2

R-IF-SF モデルの接続検出の仕組

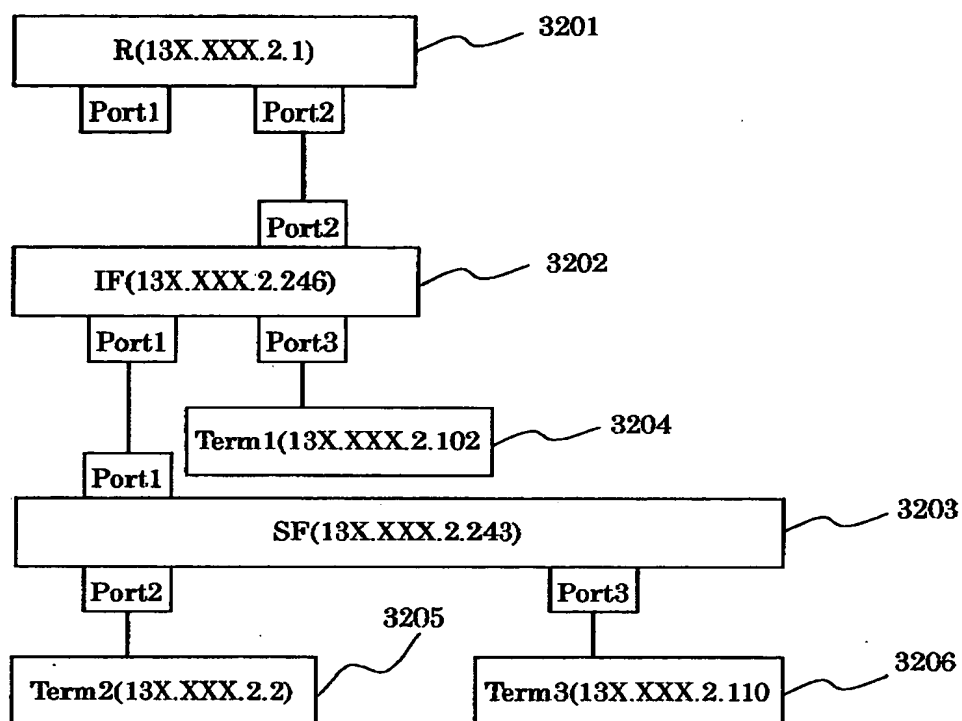


図 3 3

R-IF-SF モデルの接続検出に利用する PF テーブルのエントリ

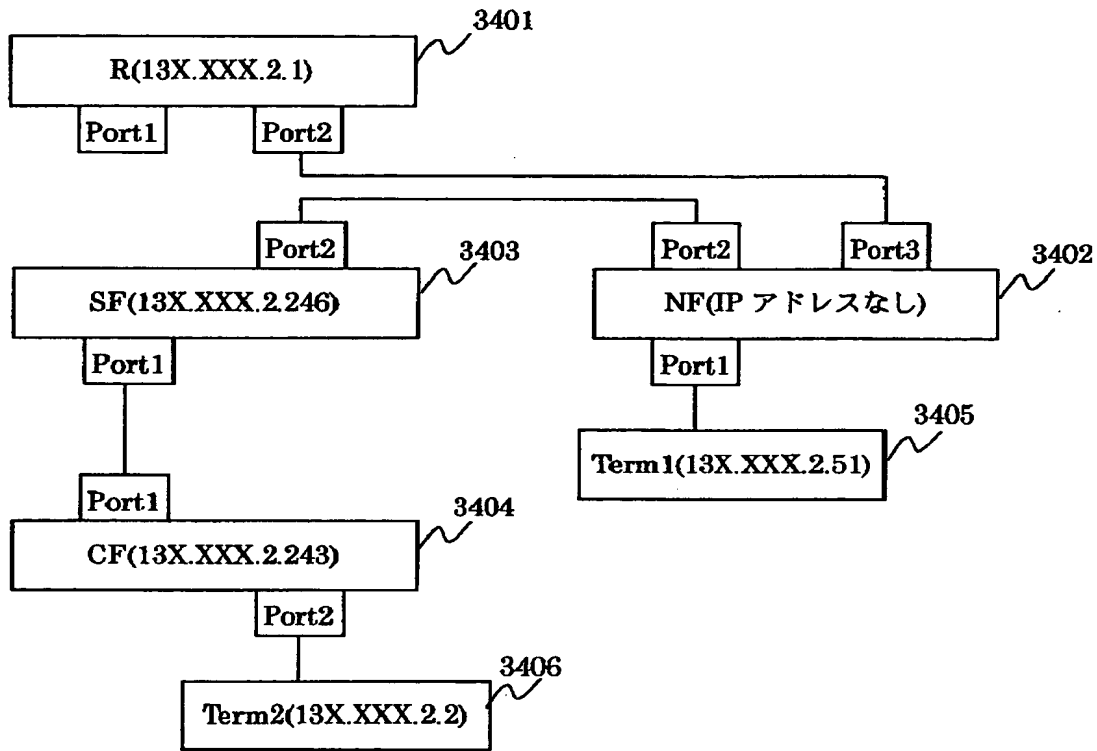
1124

Source IP Address	Source Mac Address	Source Port	Destination IP Address	Destination Mac Address
...
13X.XXX.2.246	08:00:4e:4f:ad:27	1	13X.XXX.2.2	00:e0:f7:26:a4:e8
13X.XXX.2.246	08:00:4e:4f:ad:27	1	13X.XXX.2.110	00:e0:18:00:3a:9f
13X.XXX.2.246	08:00:4e:4f:ad:27	2	13X.XXX.2.1	00:e0:f7:26:a4:e3
13X.XXX.2.246	08:00:4e:4f:ad:27	3	13X.XXX.2.102	00:e0:18:00:27:d7
...
13X.XXX.2.243	00:00:f4:71:01:37	1	13X.XXX.2.102	00:e0:18:00:27:d7
13X.XXX.2.243	00:00:f4:71:01:37	2	13X.XXX.2.2	00:e0:f7:26:a4:e3
13X.XXX.2.243	00:00:f4:71:01:37	3	13X.XXX.2.110	00:e0:18:00:3a:9f
...

【図 3 4】

図 3 4

R-SF-CF モデルの接続検出の仕組



【図 3 5】

図 3 5

R-SF-CF モデルの接続検出に利用する PF テーブルのエントリ

1124

Source IP Address	Source Mac Address	Source Port	Destination IP Address	Destination Mac Address
...
13X.XXX.2.246	08:00:4e:4f:ad:27	1	13X.XXX.2.2	00:e0:f7:26:a4:e3
13X.XXX.2.246	08:00:4e:4f:ad:27	2	13X.XXX.2.51	00:00:92:96:b4:43
...
13X.XXX.2.243	00:00:f4:71:01:37	1	13X.XXX.2.1	00:e0:f7:26:a4:e3
13X.XXX.2.243	00:00:f4:71:01:37	1	13X.XXX.2.51	00:00:92:96:b4:43
13X.XXX.2.243	00:00:f4:71:01:37	1	13X.XXX.2.246	08:00:4e:4f:ad:27
13X.XXX.2.243	00:00:f4:71:01:37	2	13X.XXX.2.2	00:e0:f7:26:a4:e3
...

3501

3502

3503

3504

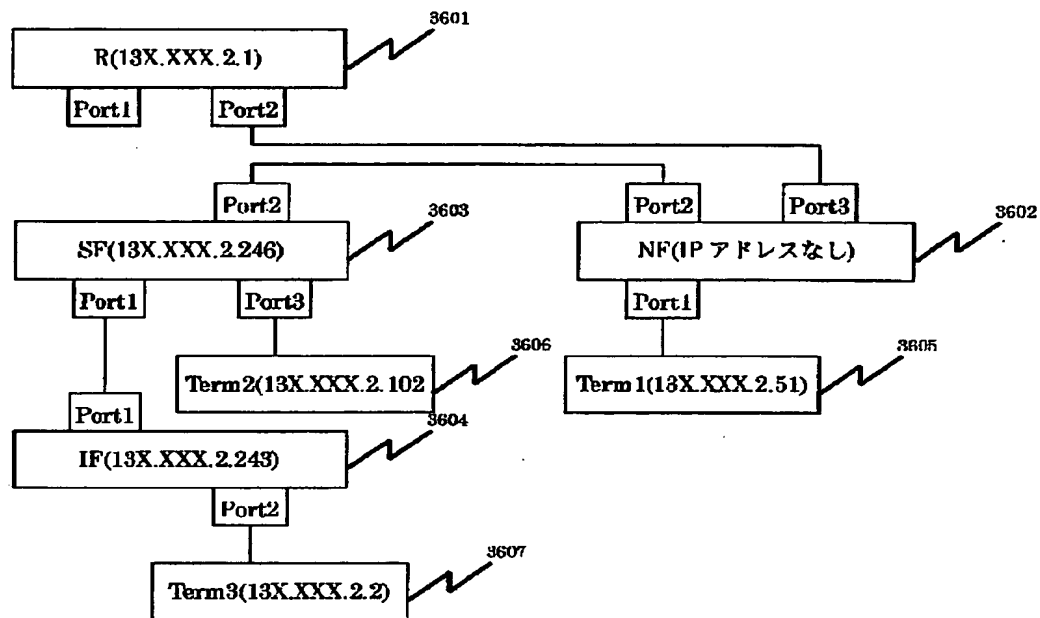
3505

3506

【図 3 6】

図 3 6


R-SF-IF モデルの接続検出の仕組



【図 37】

図 37

1124



R-SF-IF モデルの接続検出に利用する PF テーブルのエントリ

Source IP Address	Source Mac Address	Source Port	Destination IP Address	Destination Mac Address
...
13X.XX.2.246	08:00:4e:4f:ad:27	1	13X.XXX.2.2	00:e0:f7:26:a4:e3
13X.XXX.2.246	08:00:4e:4f:ad:27	2	13X.XXX.2.51	00:00:92:96:b4:43
13X.XXX.2.246	08:00:4e:4f:ad:27	3	13X.XXX.2.102	00:e0:18:00:27:d7
...
13X.XXX.2.243	00:00:f4:71:01:37	1	13X.XXX.2.1	00:e0:f7:26:a4:e3
13X.XXX.2.243	00:00:f4:71:01:37	1	13X.XXX.2.51	00:00:92:96:b4:43
13X.XXX.2.243	00:00:f4:71:01:37	1	13X.XXX.2.102	00:e0:18:00:27:d7
13X.XXX.2.243	00:00:f4:71:01:37	2	13X.XXX.2.2	00:e0:f7:26:a4:e3
...

3701

3702

3703

3704

3705

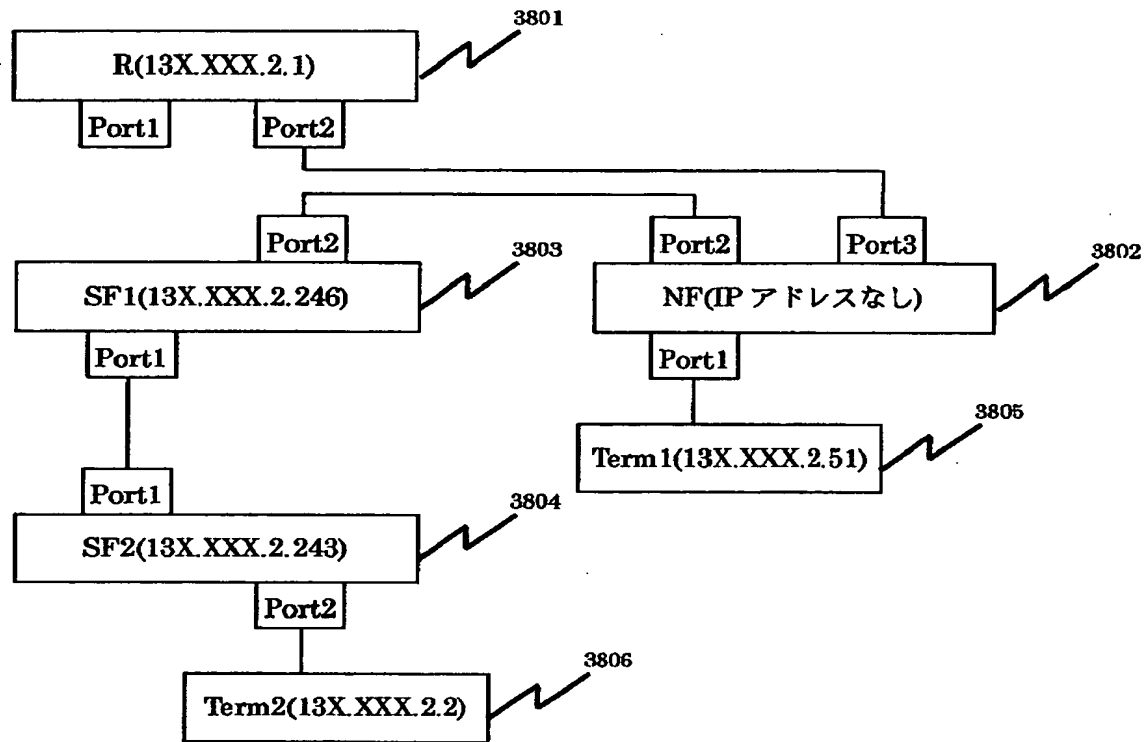
3706

3707

【図 3 8】

図 3 8

R-SF-SF モデルの接続検出の仕組



【図 3 9】

図 3 9

1124

R-SF-SF モデルの接続検出に利用する PF テーブルのエントリ

Source IP Address	Source Mac Address	Source Port	Destination IP Address	Destination Mac Address
...
13X.XXX.2.246	08:00:4e:4f:ad:27	1	13X.XXX.2.2	00:e0:f7:26:a4:e3
13X.XXX.2.246	08:00:4e:4f:ad:27	2	13X.XXX.2.51	00:00:92:96:b4:43
...
13X.XXX.2.243	00:00:f4:71:01:37	1	13X.XXX.2.51	00:00:92:96:b4:43
13X.XXX.2.243	00:00:f4:71:01:37	2	13X.XXX.2.2	00:e0:f7:26:a4:e3
...

3901

3902

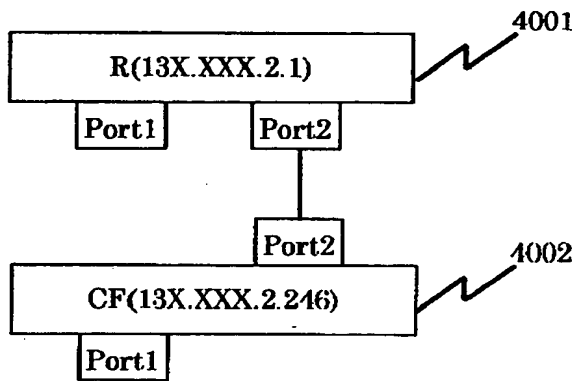
3903

3904

【図 4 0】

図 4 0

R-CF モデルの接続検出の仕組



【図 4 1】

図 4 1

1124
R-CF モデルの接続検出に利用する PF テーブルのエントリ

Source IP Address	Source Mac Address	Source Port	Destination IP Address	Destination Mac Address
...
13X.XXX.2.1	00:e0:f7:26:a4:e3	2	13X.XXX.2.246	08:00:4e:4f:ad:27
13X.XXX.2.246	08:00:4e:4f:ad:27	2	13X.XXX.2.1	00:e0:f7:26:a4:e3
...

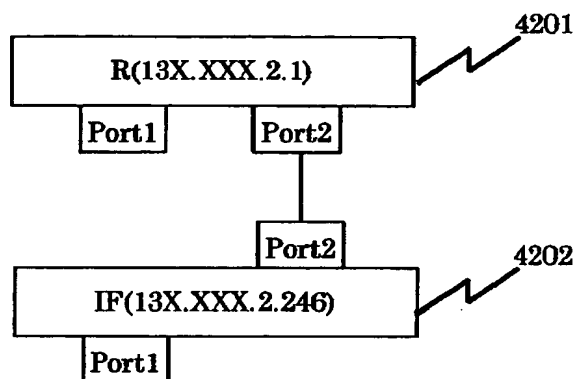
4101

4102

【図 4 2】

図 4 2

R-IF モデルの接続検出の仕組



【図 4 3】

図 4 3

1124
R-IF モデルの接続検出に利用する PF テーブルのエントリ

Source IP Address	Source Mac Address	Source Port	Destination IP Address	Destination Mac Address
...
13X.XXX.2.1	00:e0:f7:26:a4:e3	2	13X.XXX.2.246	08:00:4e:4f:ad:27
13X.XXX.2.246	08:00:4e:4f:ad:27	2	13X.XXX.2.1	00:e0:f7:26:a4:e3
...

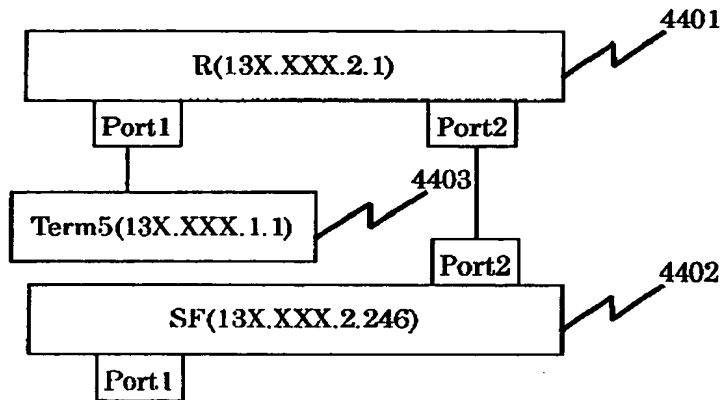
4301

4302

【図 4 4】

図 4 4

R-SF モデルの接続検出の仕組



【図 4 5】

図 4 5

R-SF モデルの接続検出に利用する PF テーブルのエントリ

1124

Source IP Address	Source Mac Address	Source Port	Destination IP Address	Destination Mac Address
...
13X.XXX.2.1	00:e0:f7:26:a4:e3	2	13X.XXX.2.246	08:00:4e:4f:ad:27
13X.XXX.2.246	08:00:4e:4f:ad:27	2	13X.XXX.1.1	08:00:20:74:d5:86
...

4501

4502

【図 4 6】

図 4 6

ネットワーク中継装置同上の接続検出方法

接続 モデル	親から子 への接続 ポート	子から親 への接続 ポート	親子 関係	接続検出するための条件
R-CF1-CF2	○	○	○	-
R-CF-IF	○	○	○	-
R-CF-SF	○	△	○	(1)CF から SF への接続ポート以外のポートに 1 台以上の機器が接続 (2)SF のフォワーディングテーブルに(1)の機器を格納
R-IF-CF	○	△	△	(1)CF から IF への接続ポート以外のポートに 1 台以上の機器が接続 (2)IF のフォワーディングテーブルに(1)の機器を格納
R-IF1-IF2	△	△	△	(1)IF1 から R への接続ポート以外のポートに 1 台以上の機器が接続 (2)IF2 のフォワーディングテーブル中の R を格納しているポートのエントリに(1)の機器を格納 (3)IF2 から R への接続ポート以外のポートに 1 台以上の機器が接続 (4)IF1 のフォワーディングテーブル中の R を格納しているポート以外のポートのエントリに(3)の機器を格納
R-IF-SF	△	△	△	(1)IF から R への接続ポート以外のポートに 2 台以上の機器が接続 (2)SF のフォワーディングテーブル中の特定のポートのエントリに(1)の機器を格納 (3)SF のフォワーディングテーブル中の(2)のポート以外のポートのエントリに(1)の機器の中で(2)の機器とは別の機器を格納 (4)IF から R への接続ポート以外のポートで、(1)以外のポートに 1 台以上の機器が接続 (5)SF のフォワーディングテーブル中の特定のポートのエントリに(4)の機器を格納

注)○：接続の検出が可能

△：接続検出するための条件を満たす場合、接続検出が可能

×：接続の検出が不可能

【図 4 7】

図 4 7

ネットワーク中継装置同士の接続検出方法

接続 モデル	親から子 への接続 ポート	子から親 への接続 ポート	親子 関係	接続検出するための条件
R-SF-CF	△	○	×	(1)CF から SF への接続ポート以外のポートに 1 台以上の機器が接続 (2)SF のフォワーディングテーブル中の特定のポートのエントリに(1)の機器を格納
R-SF-IF	△	△	×	(1)IF から R への接続ポートと同一ポートに 2 台以上の機器が接続 (2)SF のフォワーディングテーブル中の特定のポートのエントリに(1)の機器を格納 (3)SF のフォワーディングテーブル中の(2)のポート以外のポートのエントリに(1)の機器の中で(2)の機器とは別の機器を格納 (4)IF から R への接続ポート以外のポート 1 台以上の機器が接続 (5)SF のフォワーディングテーブル中の特定のポートのエントリに(4)の機器を格納
R-SF1-SF2	×	×	×	-
R-CF	△	○	○	R のフォワーディングテーブルに内部ネットワークの IP アドレスを含むポートが存在
R-IF	△	○	○	R のフォワーディングテーブルに内部ネットワークの IP アドレスを含むポートが存在
R-SF	△	△	○	(1)R のフォワーディングテーブルに内部ネットワークの IP アドレスを含むポートが存在 (2)SF のフォワーディングテーブルにバックボーンネットワークの IP アドレスを含むポートが存在

注)○：接続の検出が可能

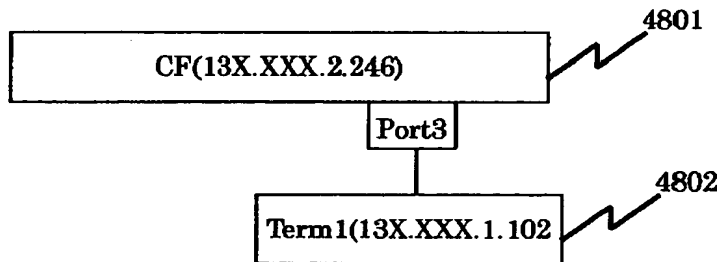
△：接続検出するための条件を満たす場合、接続検出が可能

×：接続の検出が不可能

【図 4 8】

図 4 8

CF-Term モデルの接続検出の仕組



【図 4 9】

図 4 9

CF-Term モデルの接続検出に利用する PF テーブルのエントリ

1124

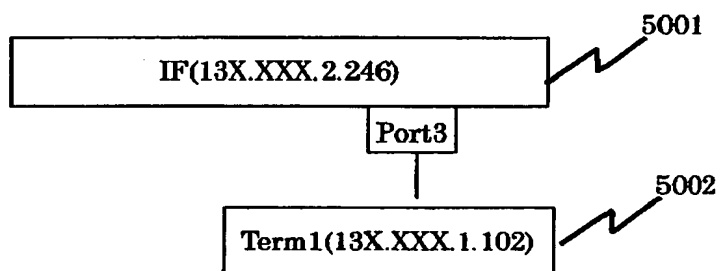
4901

Source IP Address	Source Mac Address	Source Port	Destination IP Address	Destination Mac Address
...
13X.XXX.2.246	08:00:4e:4f:ad:27	3	13X.XXX.2.102	00:e0:18:00:27:d7
...

【図 5 0】

図 5 0

IF-Term モデルの接続検出の仕組



【図 5 1】

図 5 1

1124

IF-Term モデルの接続検出に利用する PF テーブルのエントリのエントリ

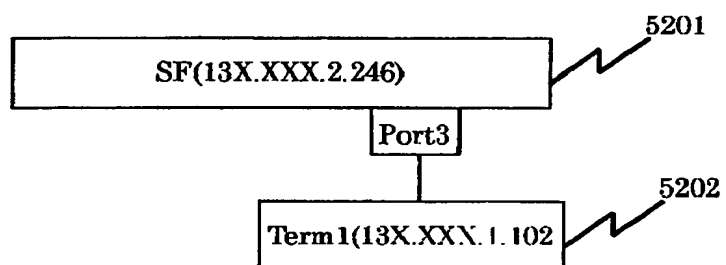
5101

Source IP Address	Source Mac Address	Source Port	Destination IP Address	Destination Mac Address
...
13X.XXX.2.246	08:00:4e:4f:ad:27	3	13X.XXX.2.102	00:e0:18:00:27:d7
...

【図 5 2】

図 5 2

SF-Term モデルの接続検出の仕組



【図 5 3】

図 5 3

1124

SF-Term モデルの接続検出に利用する PF テーブルのエントリ

Source IP Address	Source Mac Address	Source Port	Destination IP Address	Destination Mac Address
...
13X.XXX.2.246	08:00:4e:4f:ad:27	3	13X.XXX.2.102	00:e0:18:00:27:d7
...

5301

【図 5 4】

図 5 4

ネットワーク中継装置と端末装置の接続検出方法

中継装置の 接続分類	端末装置 の接続検出	接続検出するための条件
CF-Term	○	-
IF-Term	○	-
SF-Term	△	各ポートに 1 台の端末装置が接続

【図 5 5】

図 5 5

複数のモデルを組合せることによる親子関係の検出
(R-CF-CF モデルと R-CF-SF モデルを組合せ、R-SF-CF モデルの親子関係を検出する例)

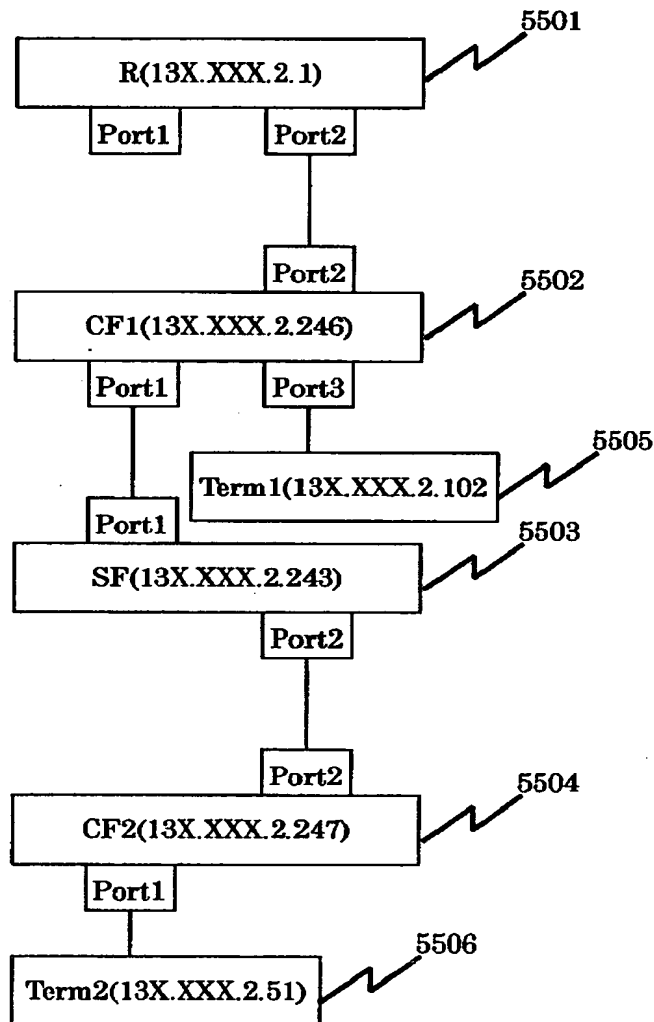


図 5 6

1125

複数のモデルを組合せることによる親子関係の検出に利用する TS テーブルのエントリ

Terminal IP Address	Terminal Mac Address	Terminal Port	Parent IP Address	Parent Mac Address	Parent Port
...
13X.XXX.2.243	00:00:f4:71:01:37	1	13X.XXX.2.246	08:00:4e:4f:ad:27	1
13X.XXX.2.247	00:00:81:39:df:aa	2	13X.XXX.2.246	08:00:4e:4f:ad:27	1
13X.XXX.2.243	00:00:f4:71:01:37	2	13X.XXX.2.247	00:00:81:39:df:aa	2
13X.XXX.2.247	00:00:81:39:df:aa	2	13X.XXX.2.243	00:00:f4:71:01:37	2
...

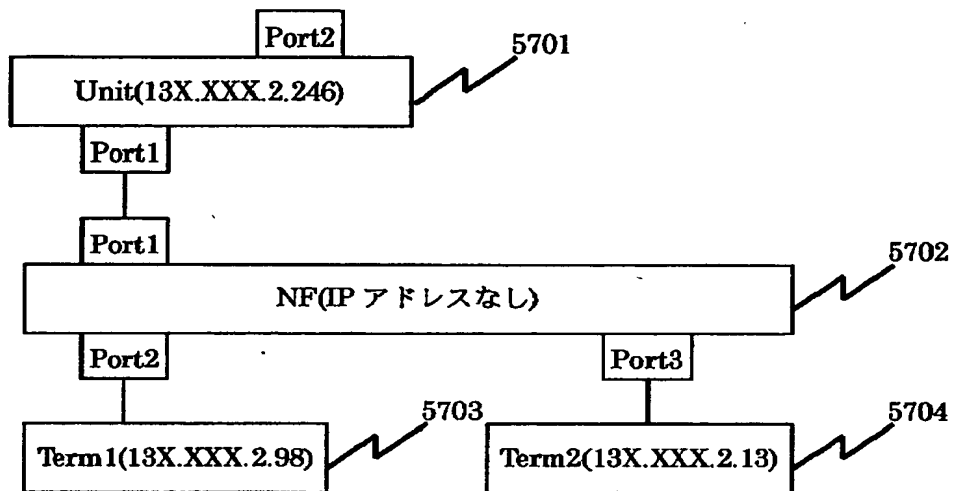
[条件]

- ① 接続関係は検出できるが、親子関係は検出できない場合、TS テーブルでは2つの対称になるエントリを格納することで表現する(13X.XXX.2.243 と 13X.XXX.2.247 はそれぞれポート 2 で接続しているが、親子関係は不明)
- ② 13X.XXX.2.243 と 13X.XXX.2.247 は 13X.XXX.2.246 の子であり、それぞれポート 1、ポート 2 で接続している
- ③ この場合、13X.XXX.2.243 は 13X.XXX.2.247 の親になる。
→ 13X.XXX.2.243 が親であると仮定すると、13X.XXX.2.243 のポート 1、ポート 2 の両方から 13X.XXX.2.246 に接続可能であり矛盾する
→ 13X.XXX.2.243 と 13X.XXX.2.247 が Non Intelligent Hub に接続し、兄弟であると仮定した場合も同様に矛盾する

【図 5 7】

図 5 7

Non Intelligent Hub の接続の予測方法



【図 58】

図 58

1125

Non Intelligent Hub の接続の予測に利用する TS テーブルのエントリ

Terminal IP Address	Terminal Mac Address	Terminal Port	Parent IP Address	Parent Mac Address	Parent Port
...
13X.XXX.2.96	00:60:97:0f:69:e4	—	13X.XXX.2.246	08:00:4e:4f:ad:27	1
13X.XXX.2.13	08:00:09:e1:51:5e	—	13X.XXX.2.246	08:00:4e:4f:ad:27	1
...

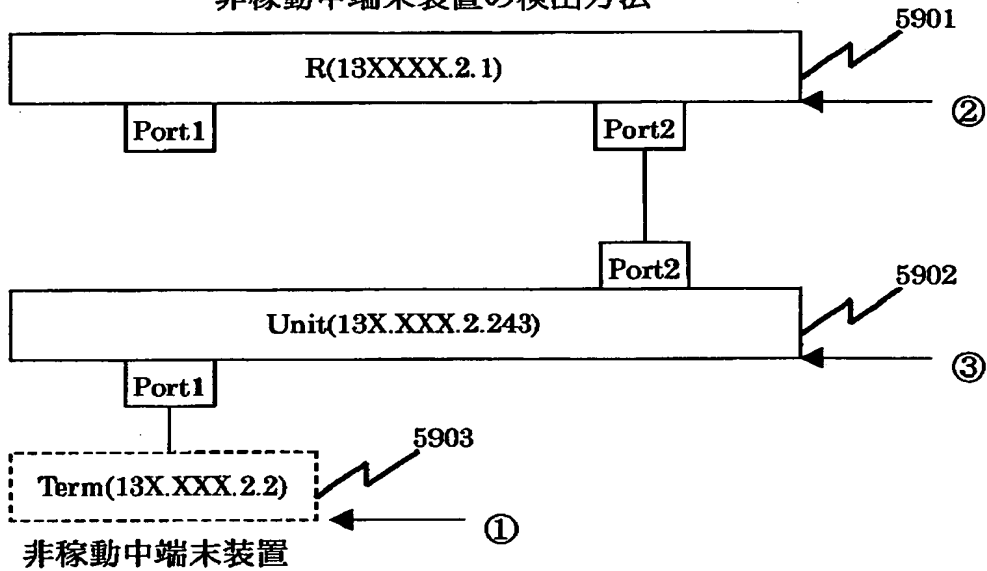
5801

5802

【図 5 9】

図 5 9

非稼動中端末装置の検出方法



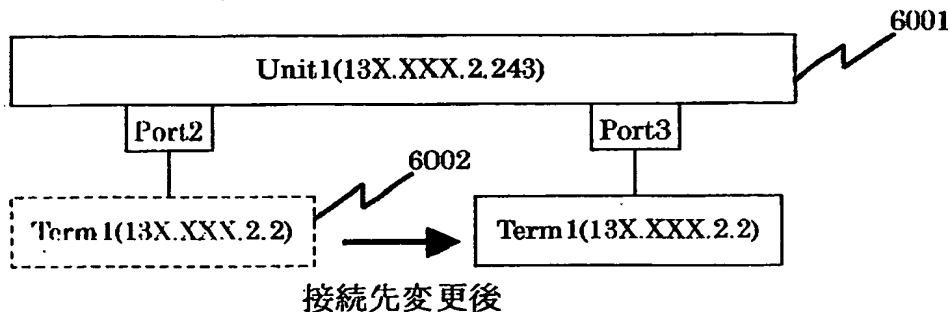
【条件】

- ① 非稼動中端末装置(133.108.2.2)がポーリングに応答せず、TI テーブルの該当するエントリの alive 値が FALSE になる。
- ② Router の ARP テーブルに非稼動中端末装置(133.108.2.2)のエントリがキャッシュされており、AT テーブルのエントリの作成が可能である。
- ③ 端末装置の接続先のネットワーク中継装置(133.108.2.243)に非稼動中端末装置(133.108.2.2)の接続情報をキャッシュしており、PF テーブルや TS テーブルのエントリの作成が可能である。

【図 6 0】

図 6 0

接続先の変更の検出方法



【図 6 1】

図 6 1

1125

接続先の変更の検出に利用する TS テーブルのエントリ

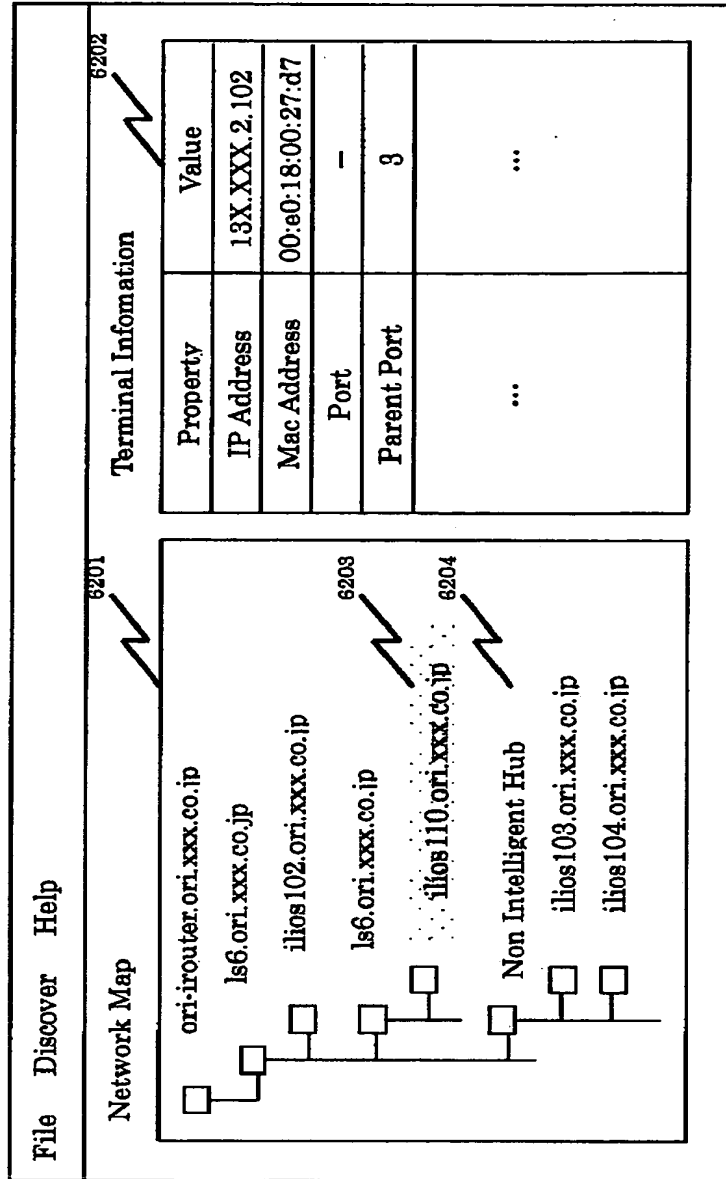
Terminal IP Address	Terminal Mac Address	Terminal Port	Parent IP Address	Parent Mac Address	Parent Port
...
13X.XXX.2.2	08:00:20:a13X:ab	-	13X.XXX.2.243	00:00:f4:71:01:37	2
...

接続先変更後

Terminal IP Address	Terminal Mac Address	Terminal Port	Parent IP Address	Parent Mac Address	Parent Port
...
13X.XXX.2.2	08:00:20:a13X:ab	-	13X.XXX.2.243	00:00:f4:71:01:37	2
13X.XXX.2.2	08:00:20:a13X:ab	-	13X.XXX.2.243	00:00:f4:71:01:37	3
...

【図 6 2】

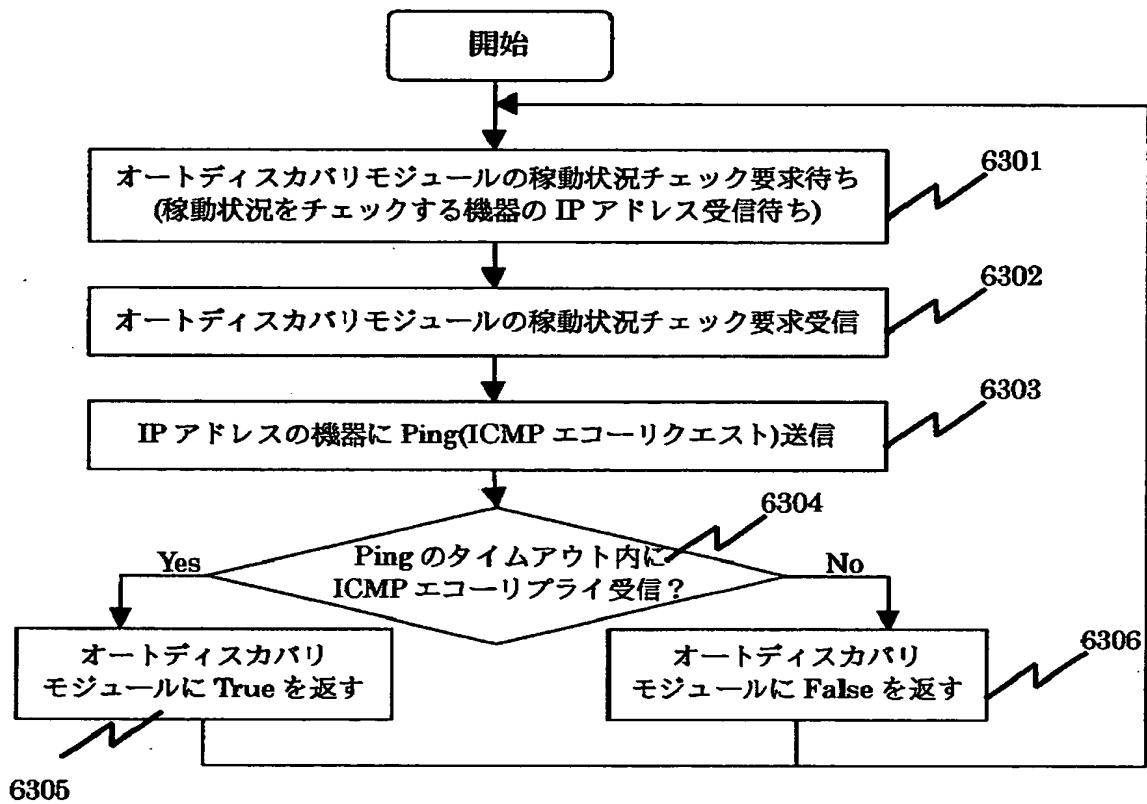
図 6 2
ネットワーク構成図面表示例



【図 6 3】

図 6 3

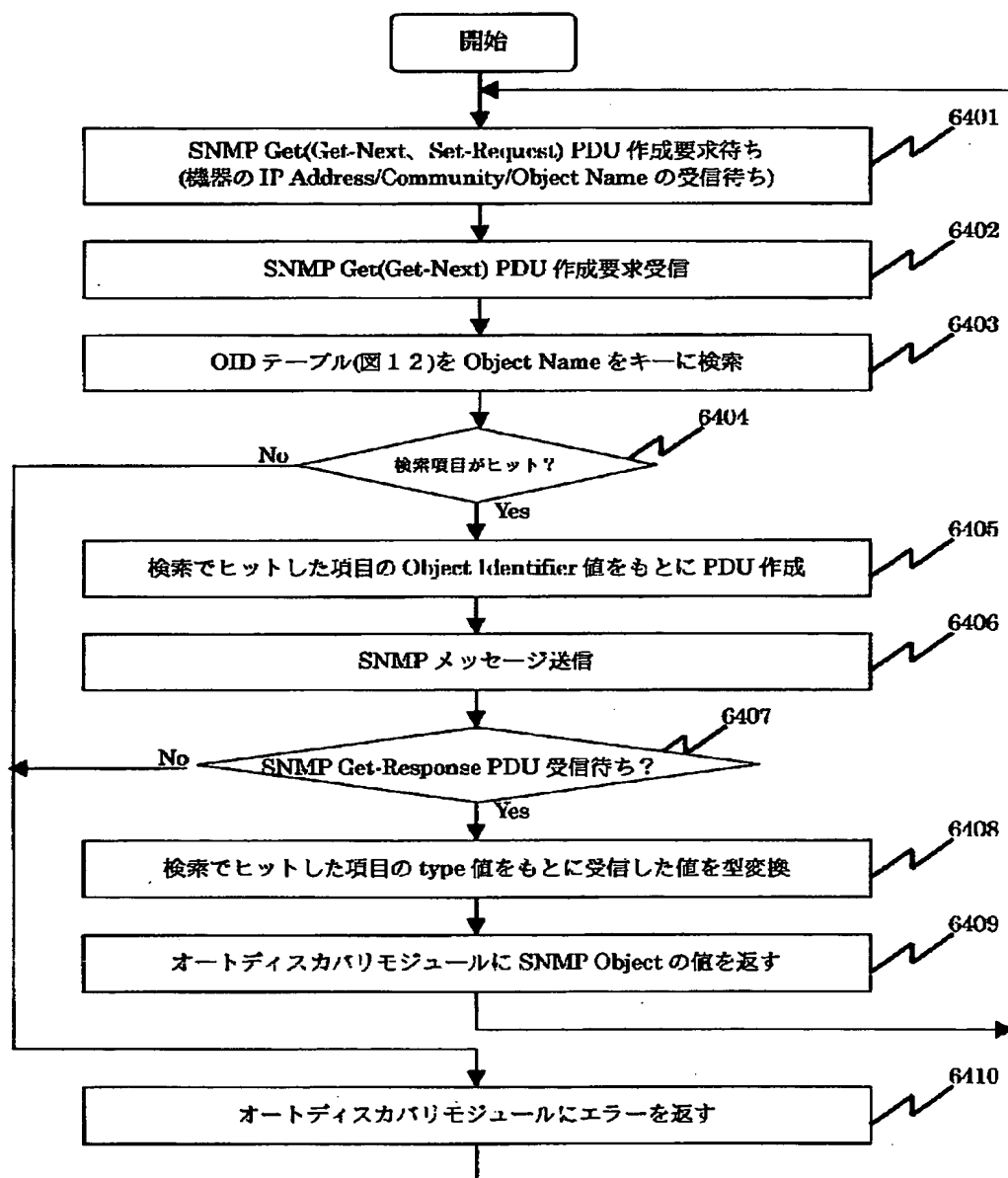
稼動状況検出モジュールの動作フローチャート
(ICMP エコーリクエストの送受信による稼動状況検出処理)



【図 6 4】

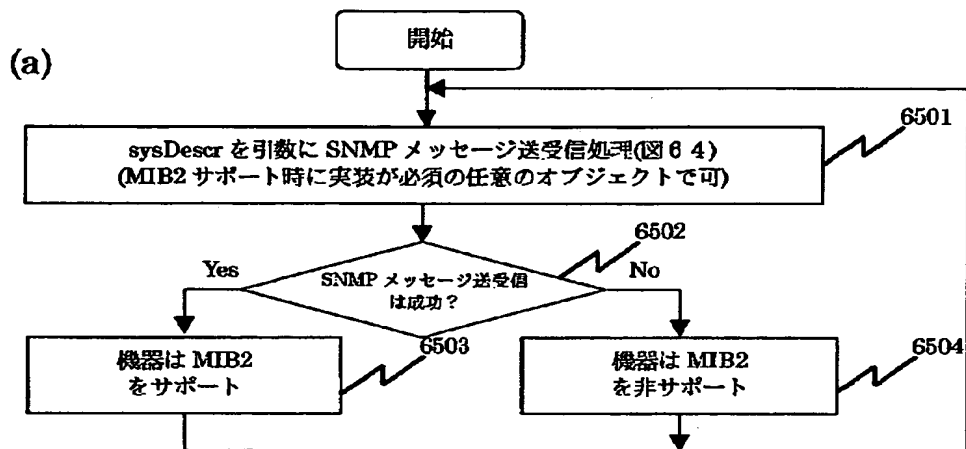
図 6 4

MIB アクセスモジュールの動作フローチャート 1
(PDU(Protocol Data Unit)を作成し、SNMP メッセージを送受信する処理)

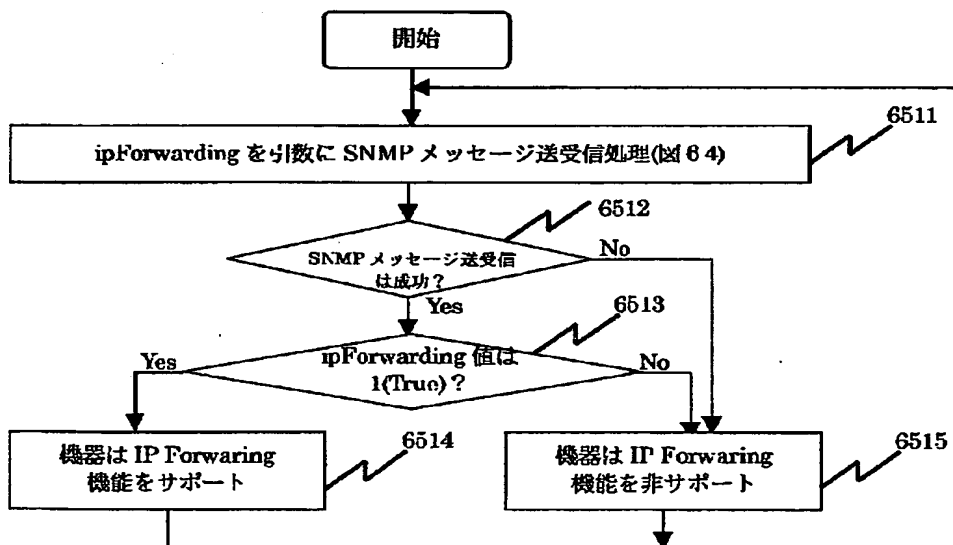


【図 65】

図 65

MIB アクセスモジュールの動作フローチャート 2
(MIB2 サポート状況チェックの処理)

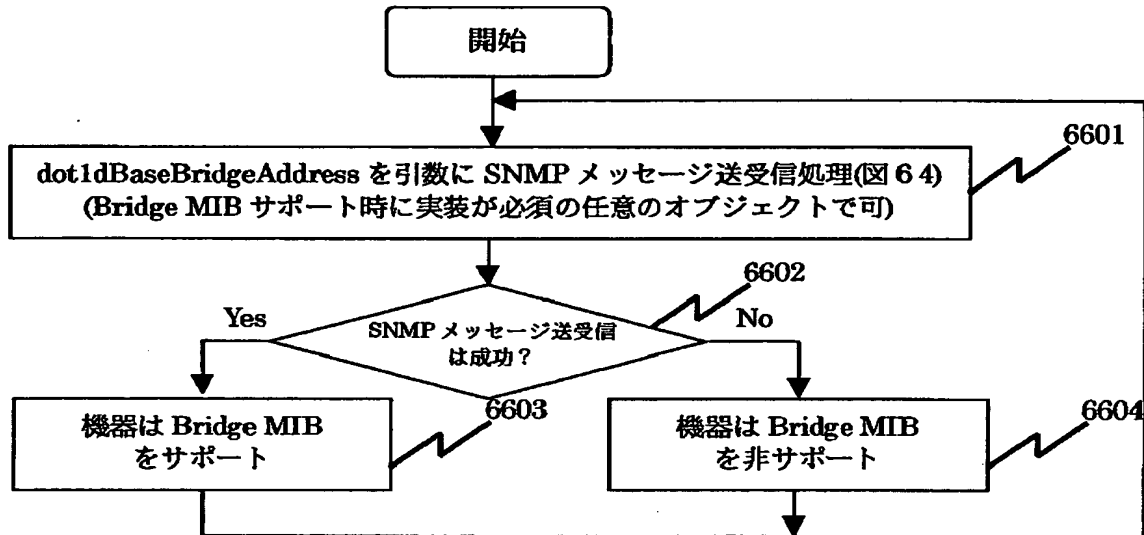
(b)

MIB アクセスモジュールの動作フローチャート 3
(IP Forwarding チェックの処理)

【図 6 6】

図 6 6

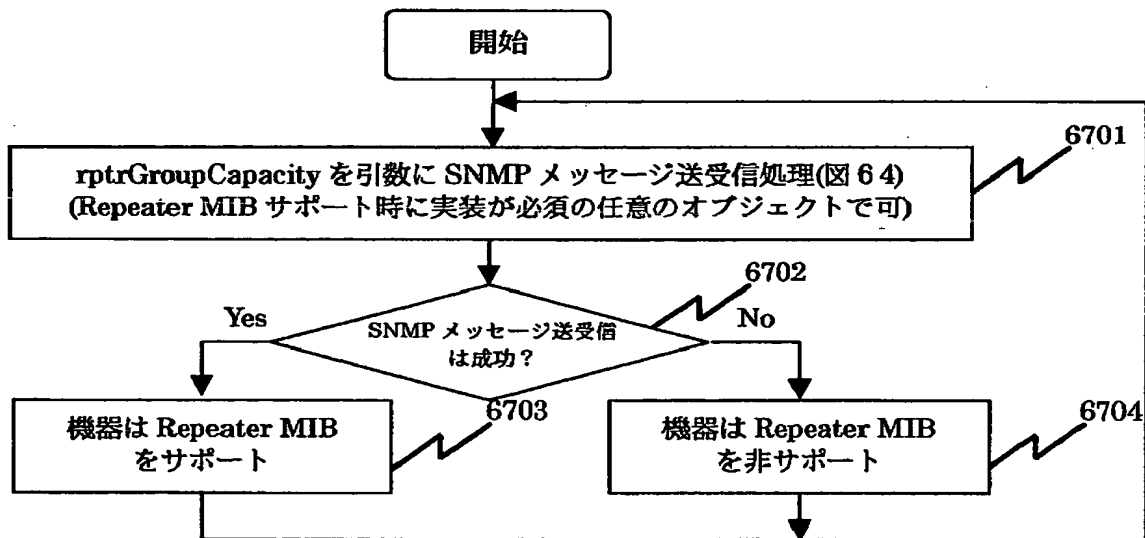
MIB アクセスモジュールの動作フローチャート 4
(Bridge MIB サポート状況チェックの処理)



【図 6 7】

図 6 7

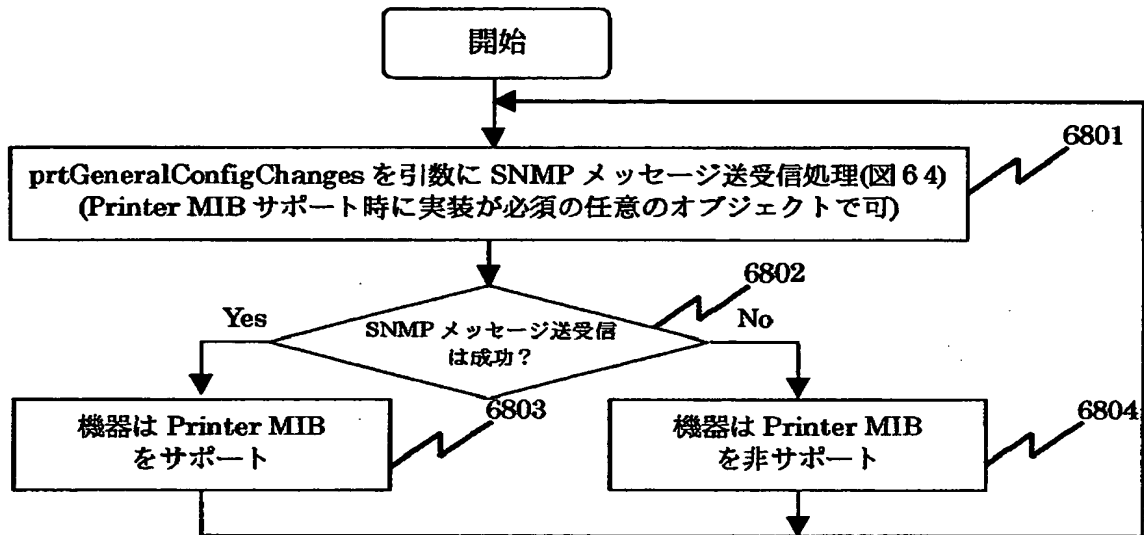
MIB アクセスモジュールの動作フローチャート 5
(Repeater MIB サポート状況チェックの処理)



【図 68】

図 68

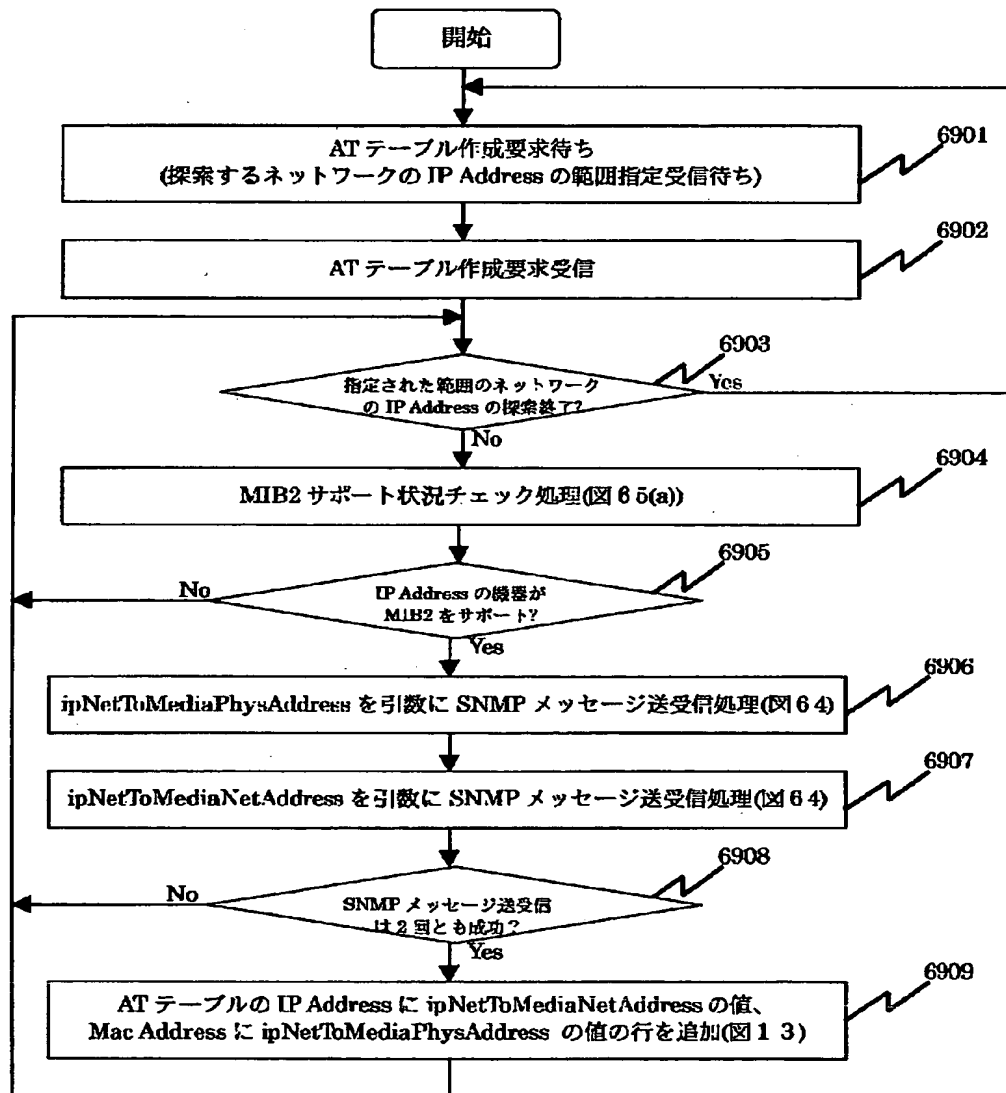
MIB アクセスモジュールの動作フローチャート 6
(Printer MIB サポート状況チェックの処理)



【図 69】

図 69

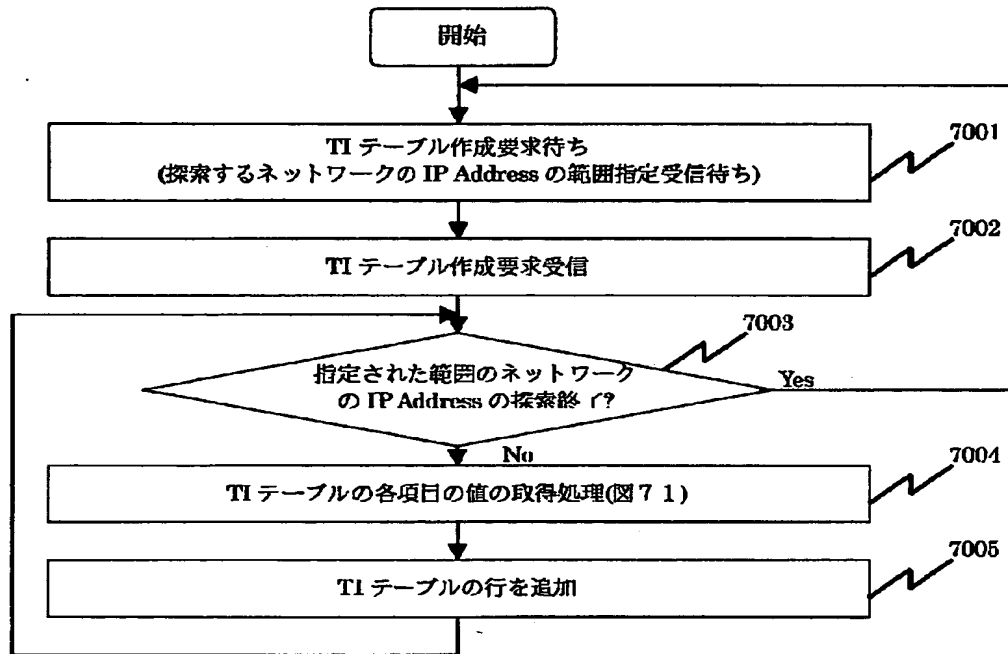
オートディスカバリモジュールの動作フローチャート 1
(AT テーブル作成の処理)



【図 7 0】

図 7 0

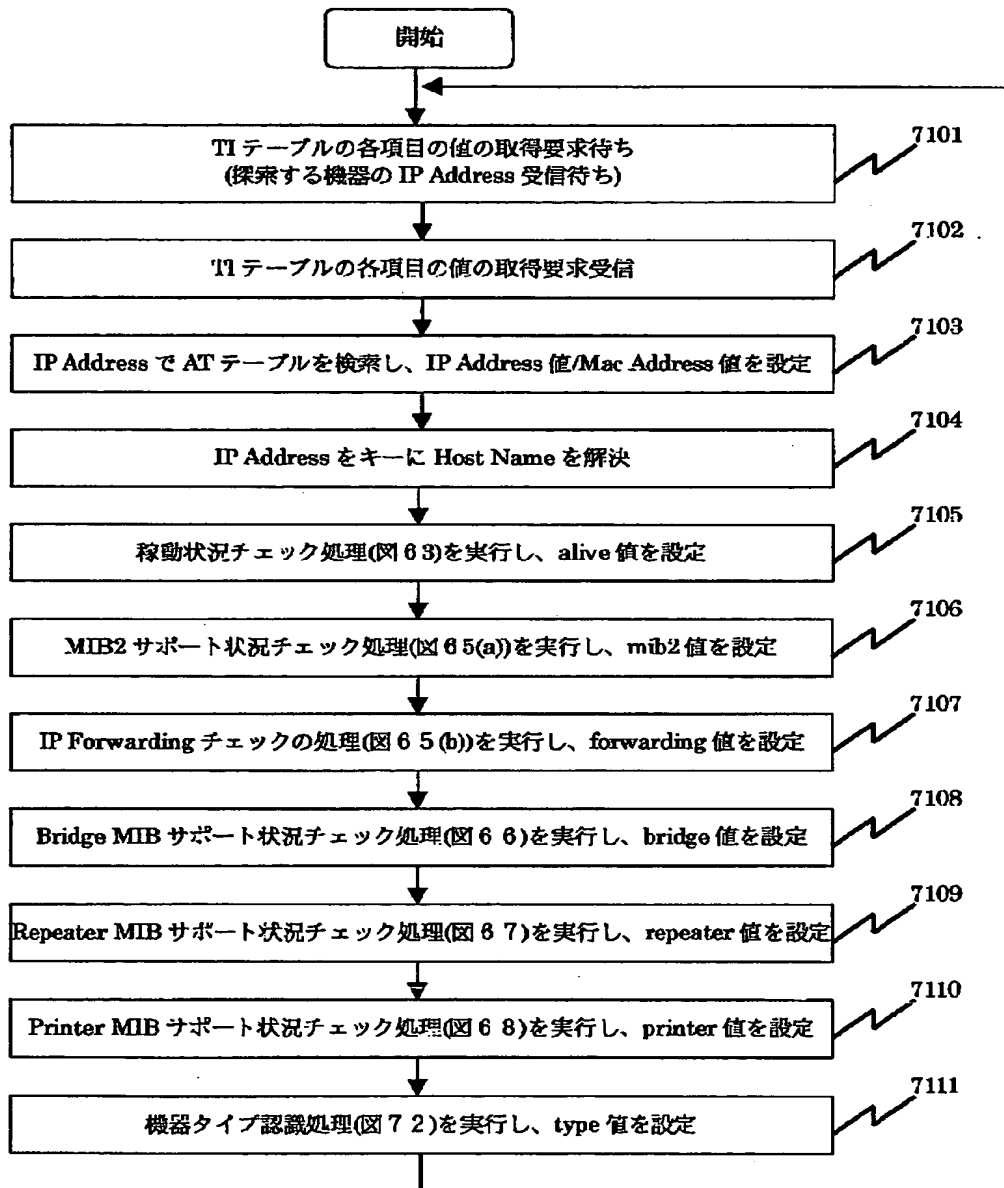
オートディスカバリモジュールの動作フローチャート 2
(TI テーブル作成の処理)



【図 7 1】

図 7 1

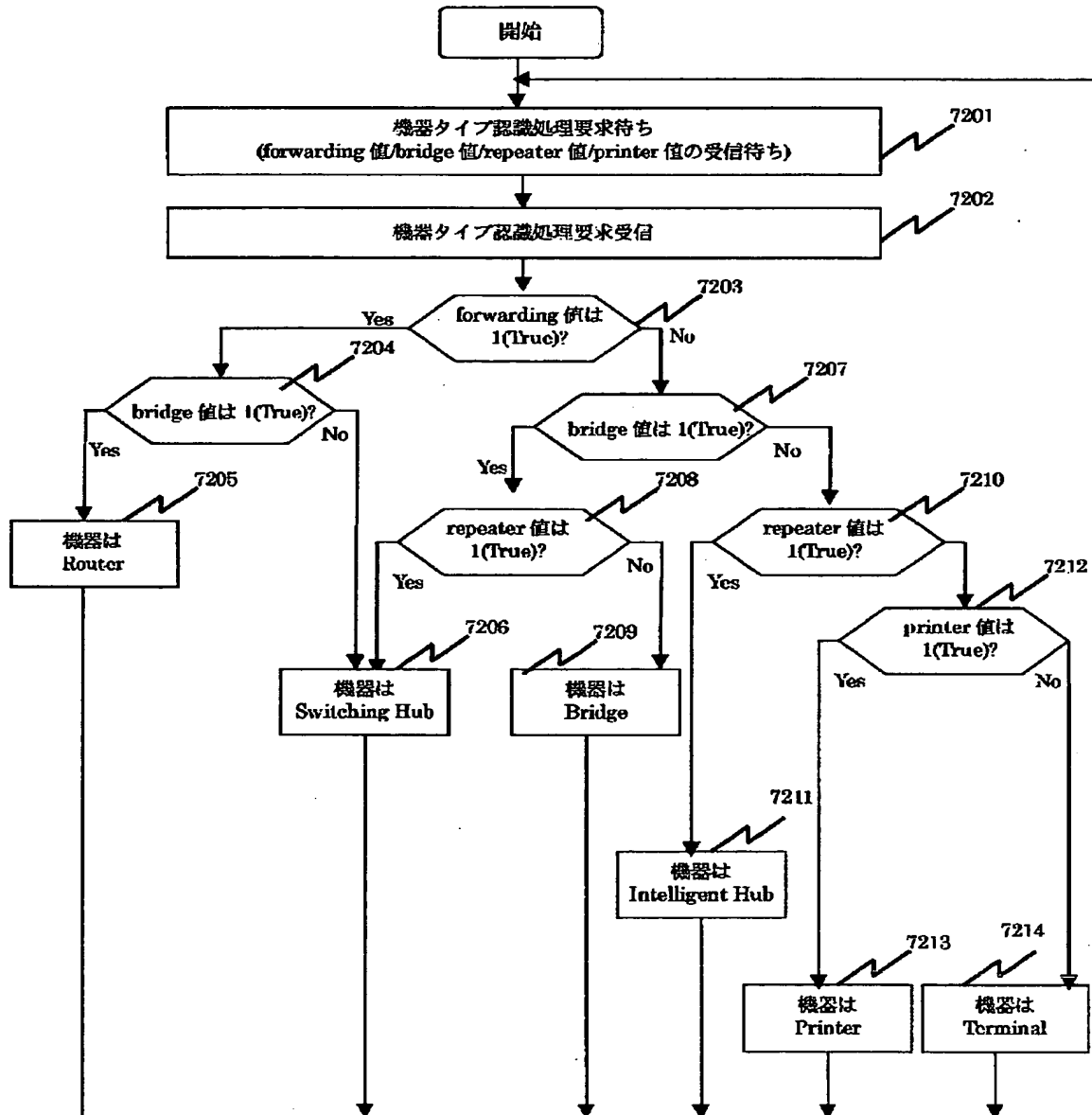
オートディスカバリモジュールの動作フローチャート 3
(TI テーブル作成(TI テーブルの各項目の値の取得処理))



【図 7 2】

図 7 2

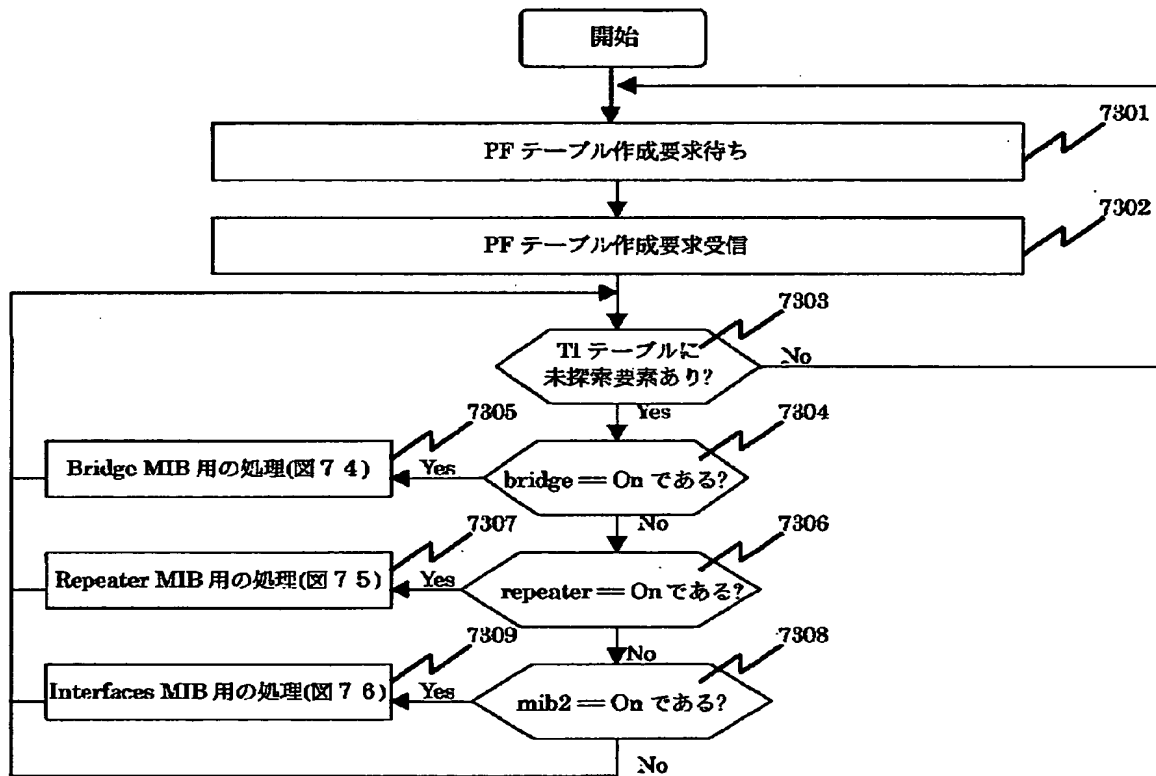
オートディスカバリモジュールの動作フローチャート 4
(TI テーブルの各項目の値の取得処理(機器タイプ認識処理(図 1 8)))



【図 7 3】

図 7 3

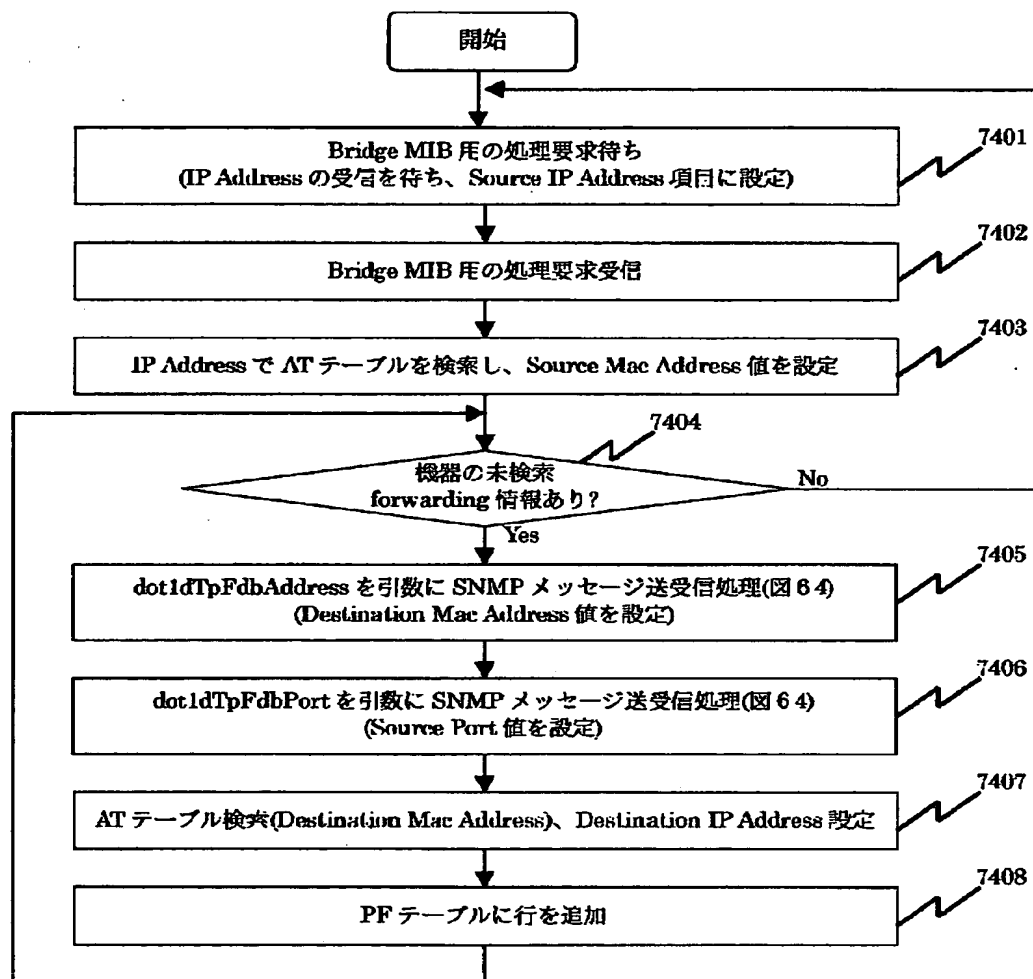
オートディスカバリモジュールの動作フローチャート 5
(PF テーブル作成の処理)



【図 7 4】

図 7 4

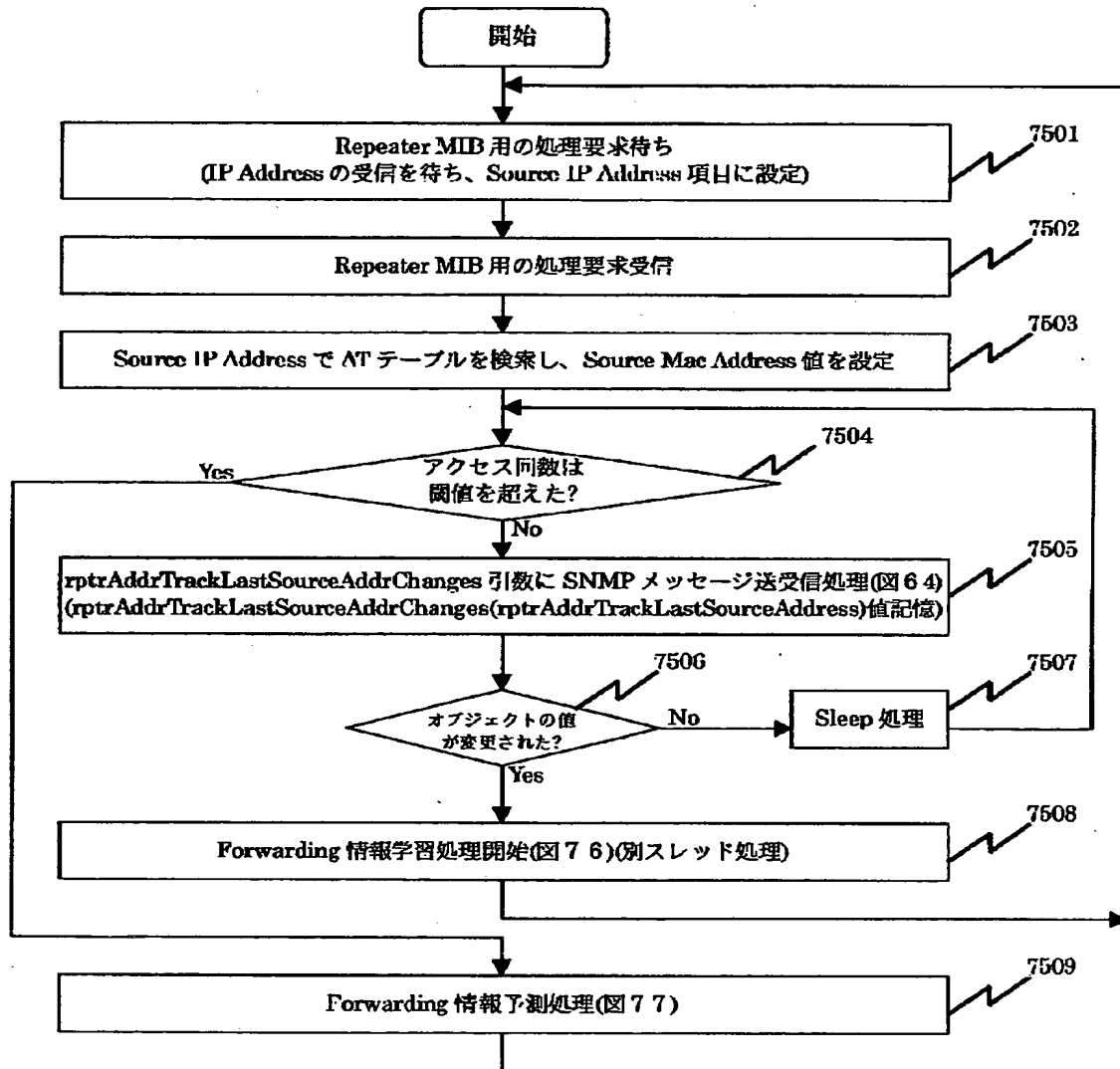
オートディスカバリモジュールの動作フローチャート 6
(PF テーブル作成(Bridge MIB 用の処理))



【図 75】

図 75

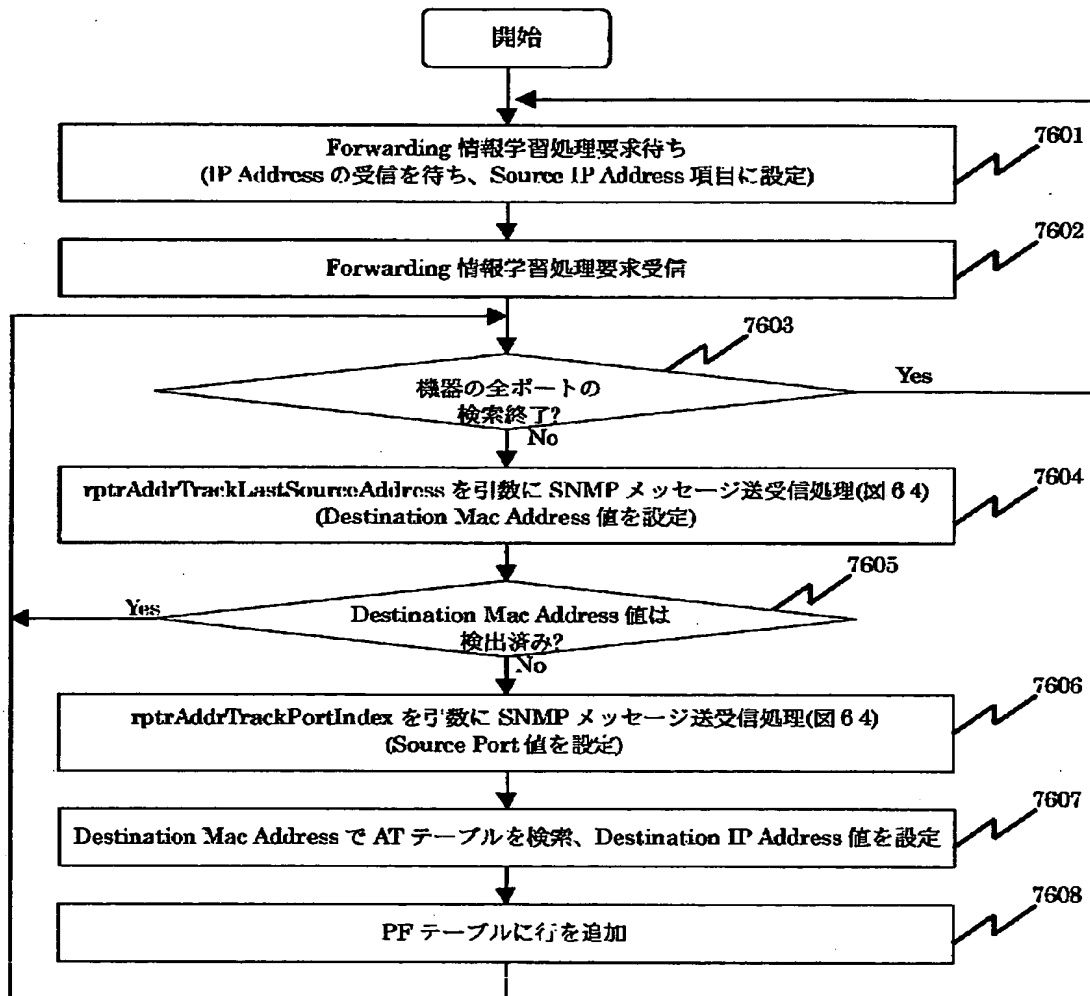
オートディスカバリモジュールの動作フローチャート 7
(PF テーブル作成(Repeater MIB 用の処理))



【図 76】

図 76

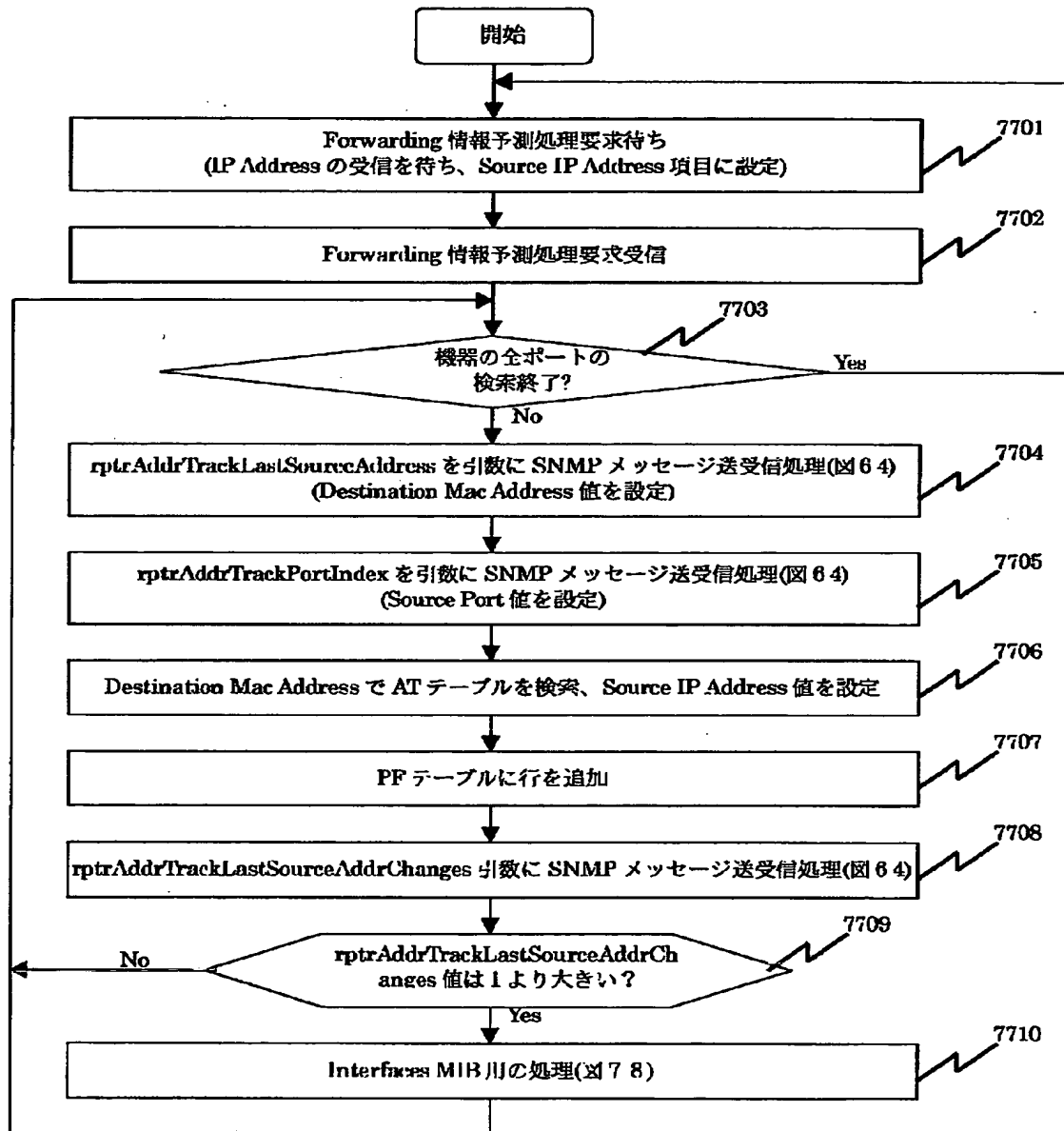
オートディスカバリモジュールの動作フローチャート 8
(Repeater MIB 用の処理(Forwarding 情報学習処理))



【図 77】

図 77

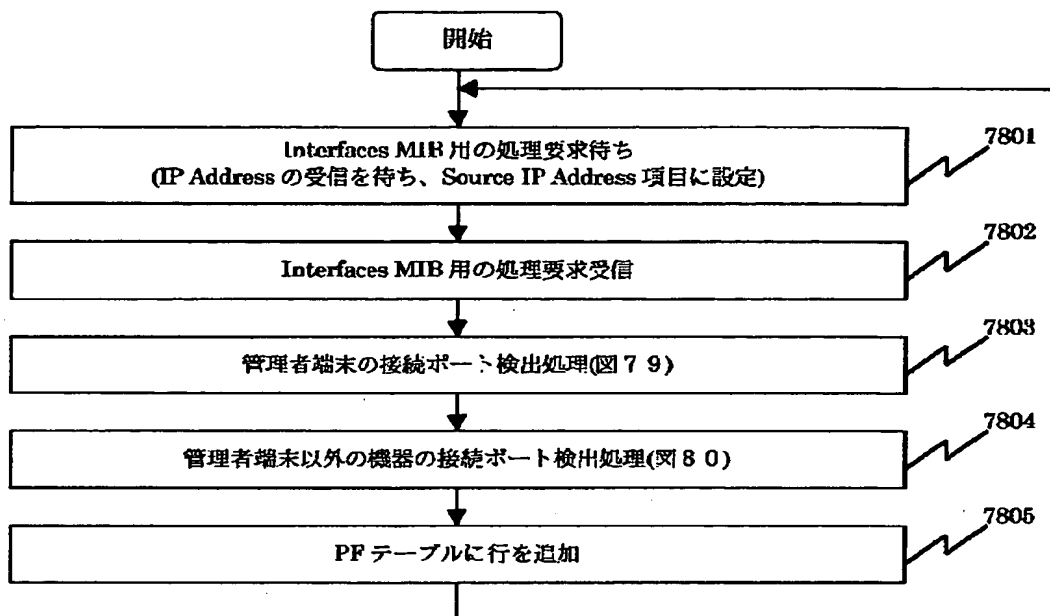
オートディスカバリモジュールの動作フローチャート 9
(Repeater MIB 用の処理(Forwarding 情報予測処理処理))



【図 7 8】

図 7 8

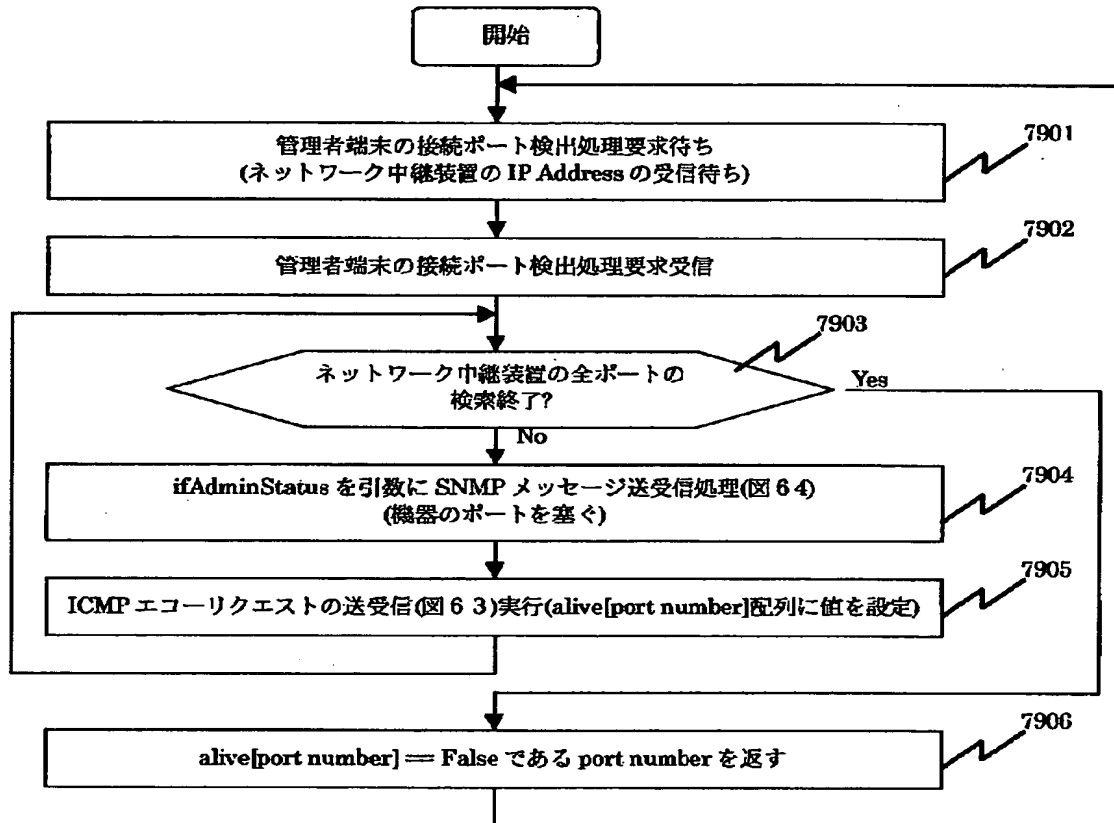
オートディスカバリモジュールの動作フローチャート 1 0
(PF テーブル作成(Interfaces MIB 用の処理))



【図 7 9】

図 7 9

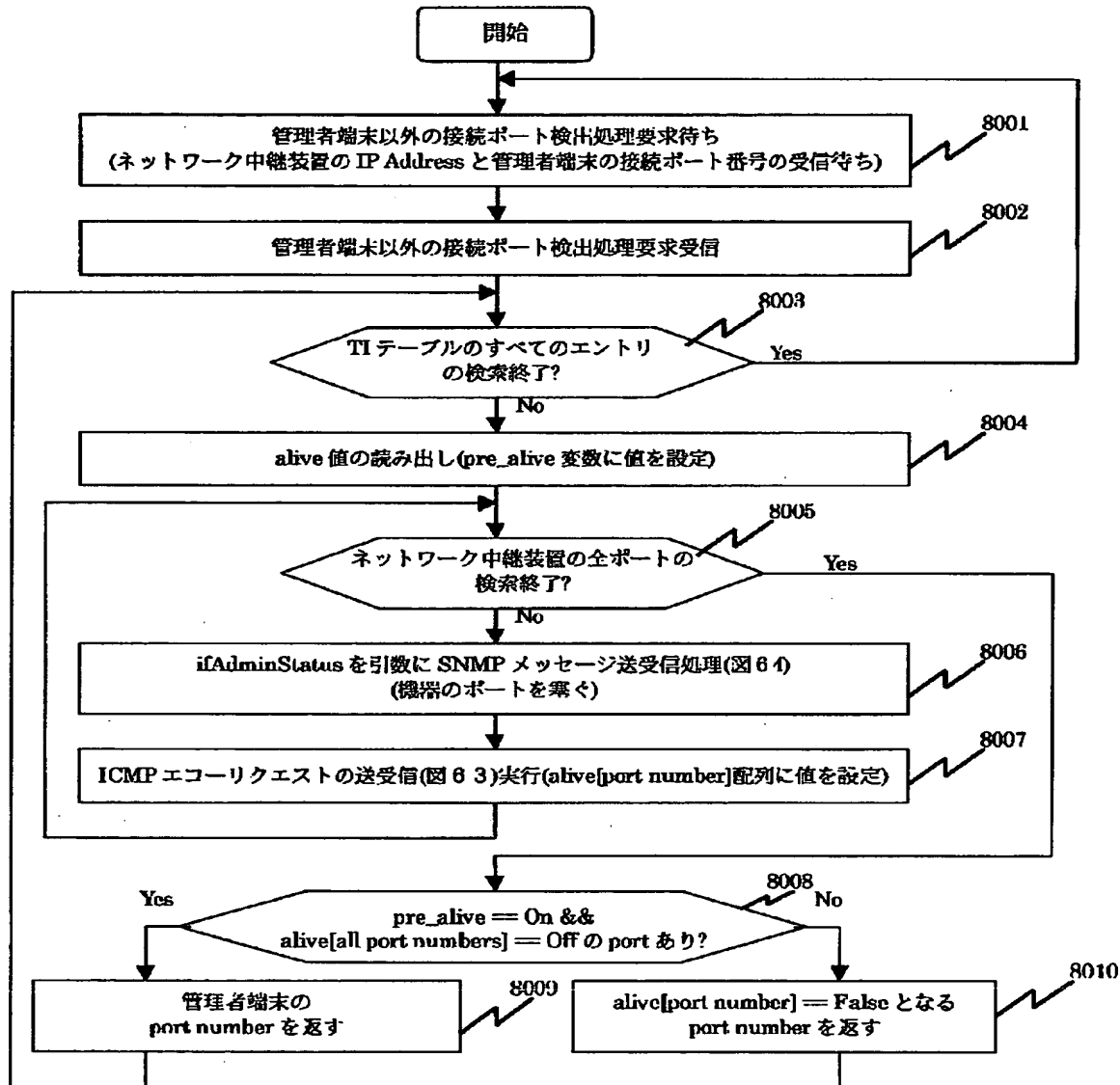
オートディスカバリモジュールの動作フローチャート 1 1
(Interfaces MIB 用の処理(管理者端末の接続ポート検出処理))



【図 80】

図 80

オートディスカバリモジュールの動作フローチャート 12
(Interfaces MIB 用の処理(管理者端末以外の機器の接続ポート検出処理))



【図 8 1】

図 8 1

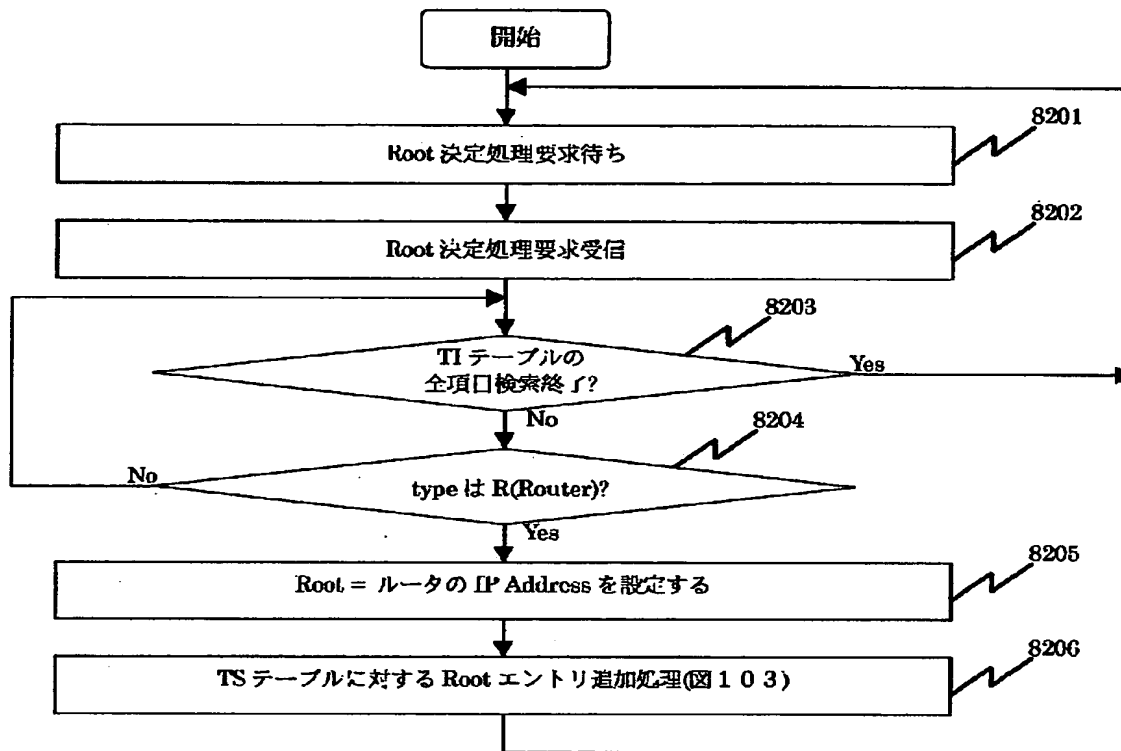
オートディスカバリモジュールの動作フローチャート 1 3
(TS テーブル作成の処理)



【図 8 2】

図 8 2

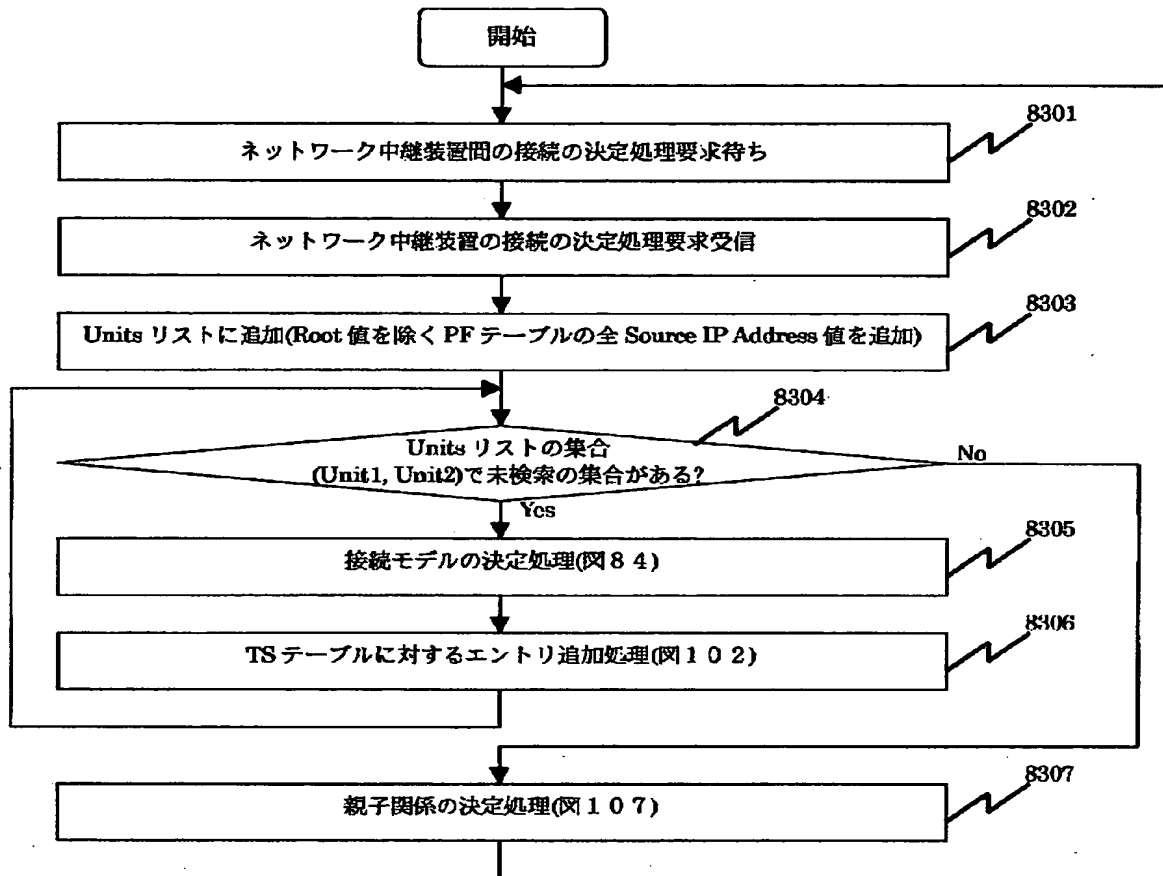
オートディスカバリモジュールの動作フローチャート 1 4
(TS テーブル作成(Root 装置決定処理))



【図 8 3】

図 8 3

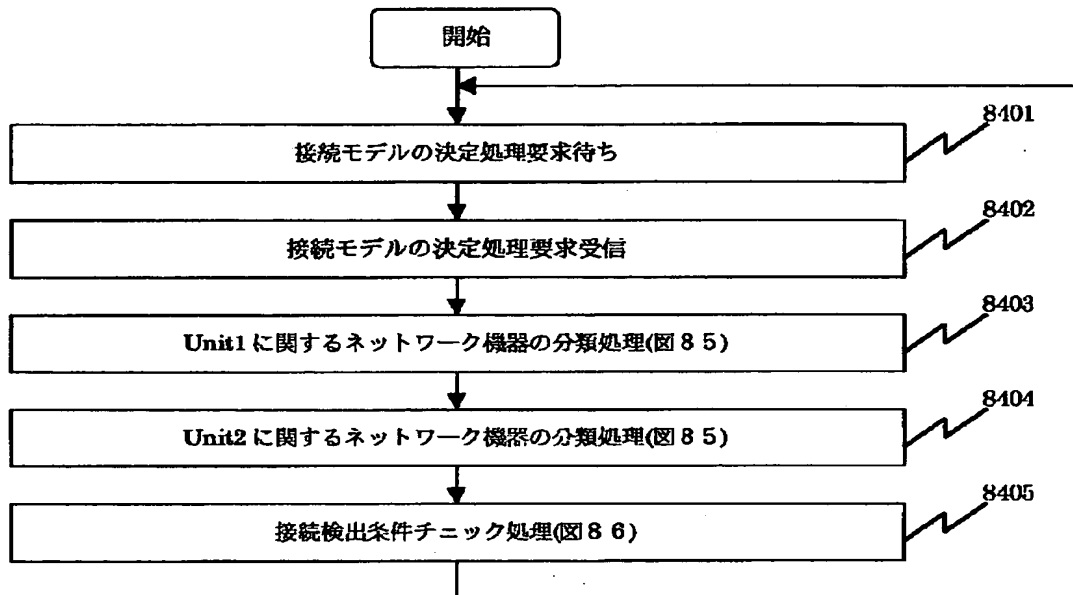
オートディスカバリモジュールの動作フローチャート 1 5
(TS テーブル作成(ネットワーク中継装置間の接続の決定処理))



【図 8 4】

図 8 4

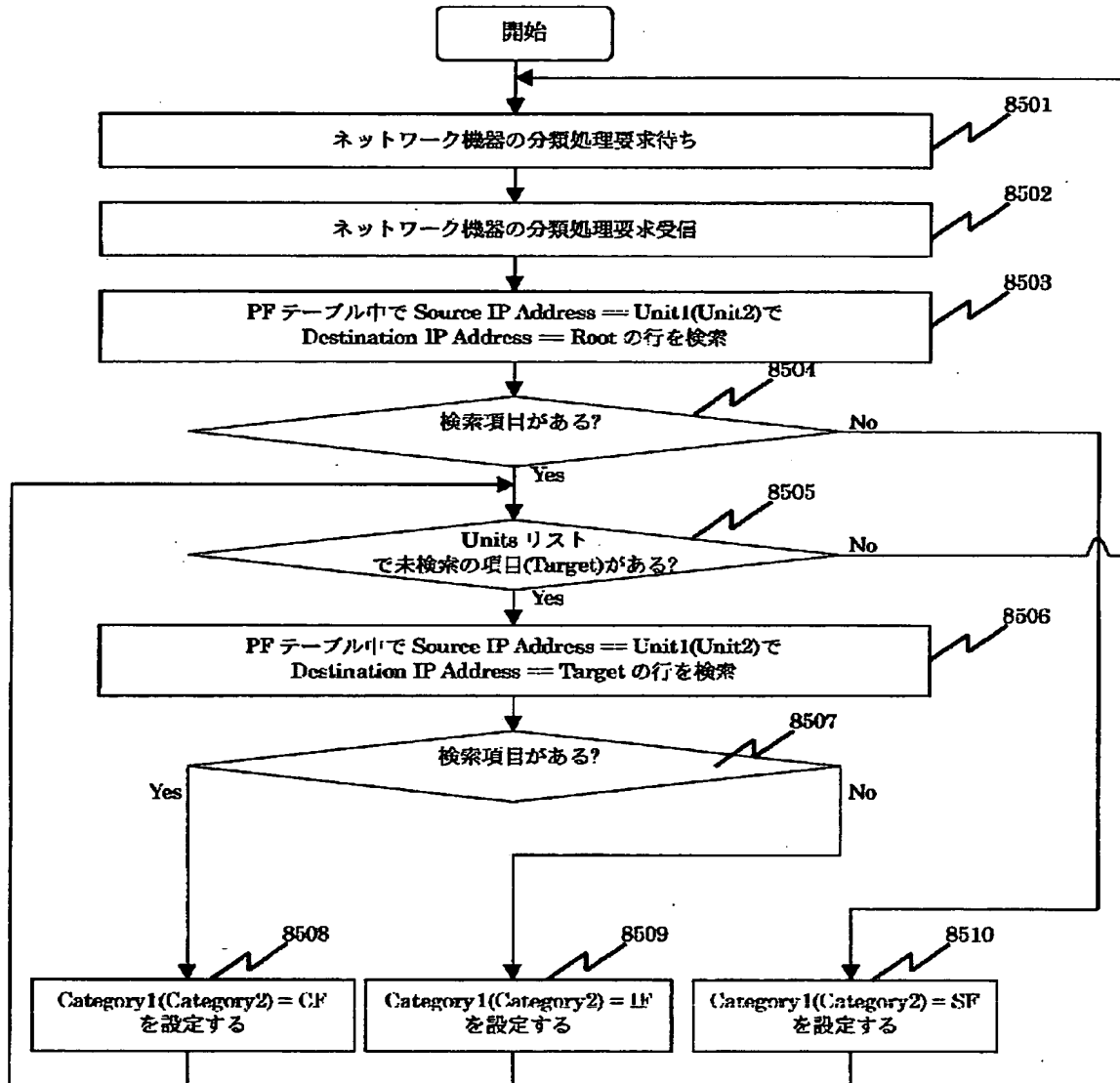
オートディスカバリモジュールの動作フローチャート 1 6
(TS テーブル作成(接続モデルの決定処理))



【図 8 5】

図 8 5

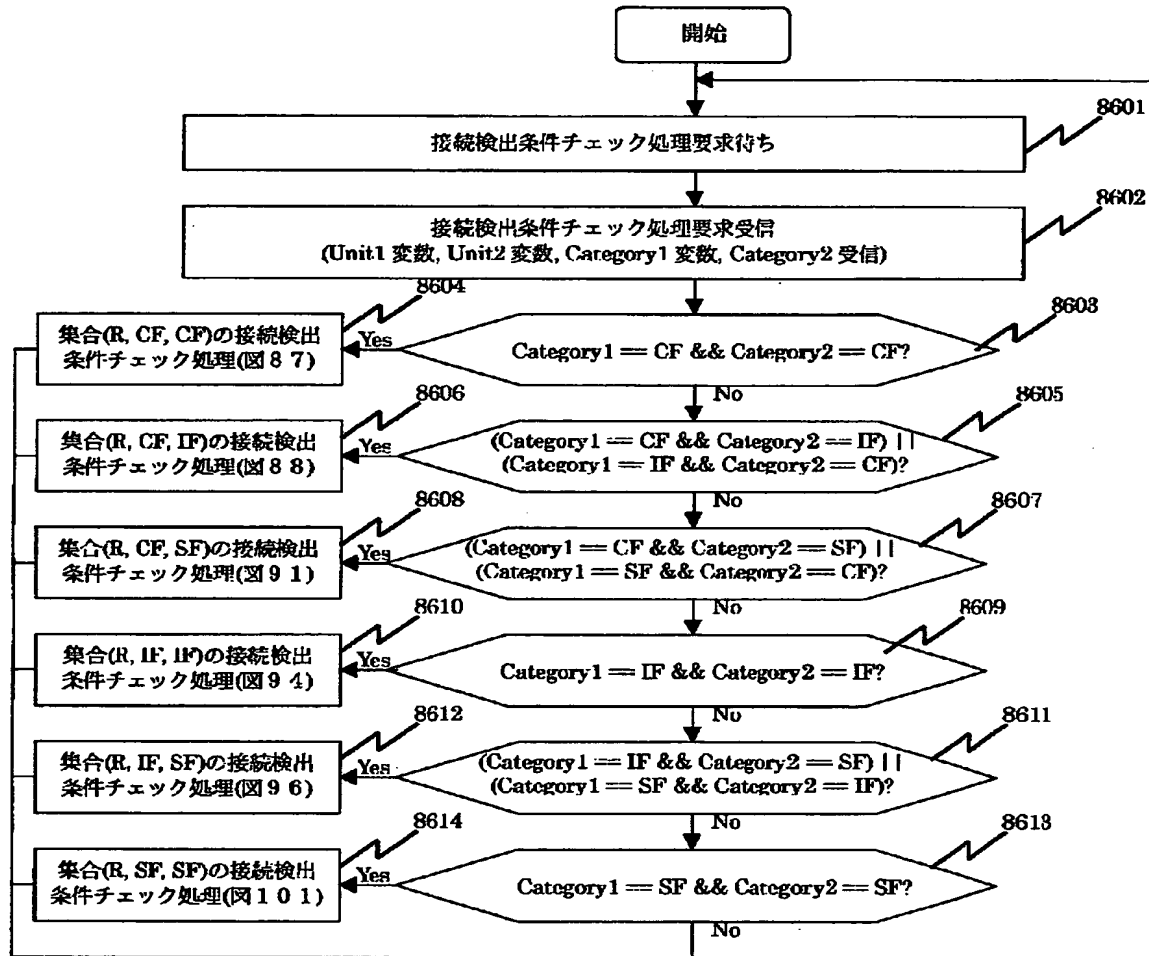
オートディスカバリモジュールの動作フローチャート 17
(TS テーブル作成(ネットワーク機器の分類処理)(図 2 1))



【図 8 6】

図 8 6

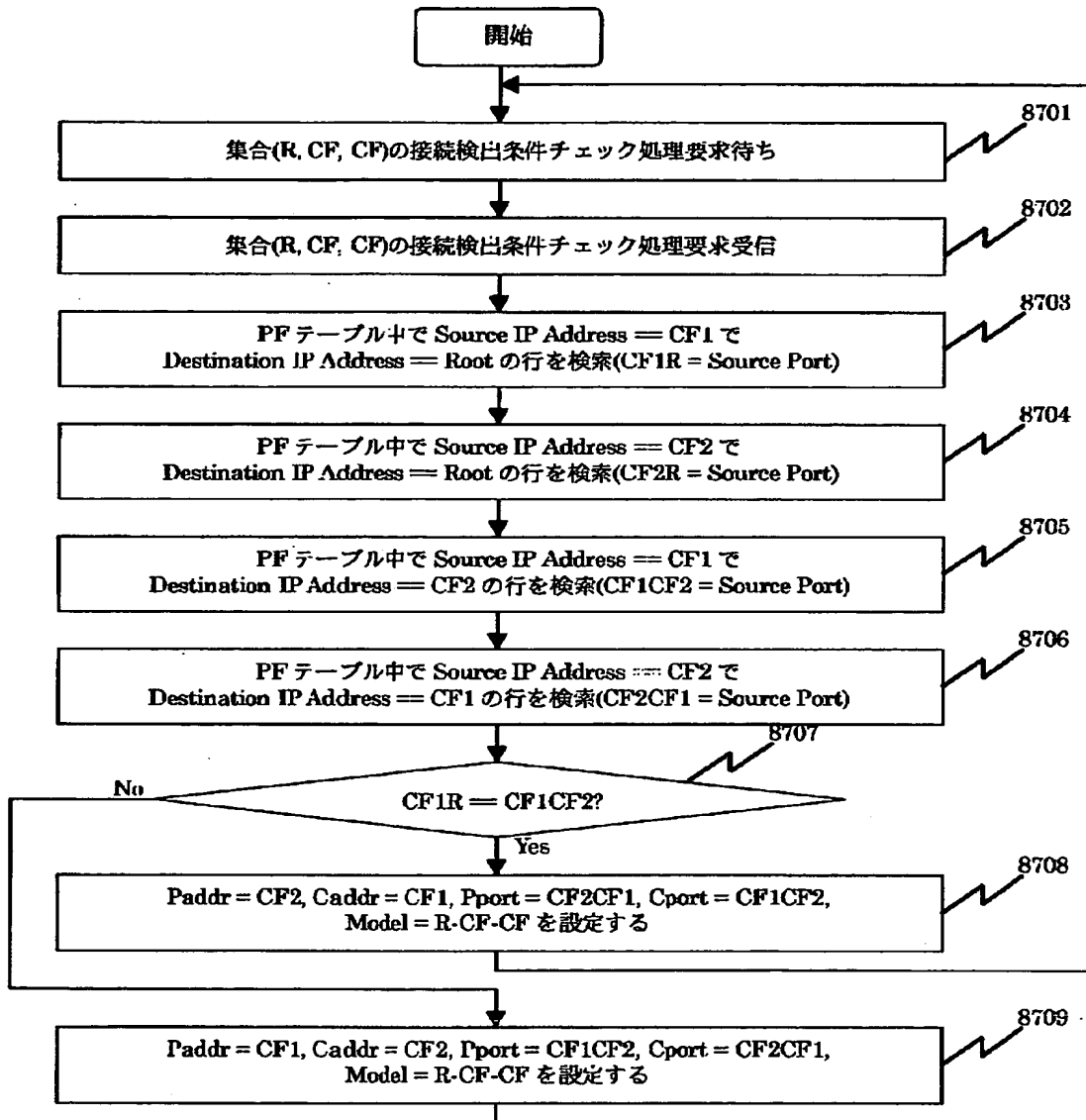
オートディスカバリモジュールの動作フローチャート 18
(TS テーブル作成(接続検出条件チェック処理)(図 4 6))



【図 8 7】

図 8 7

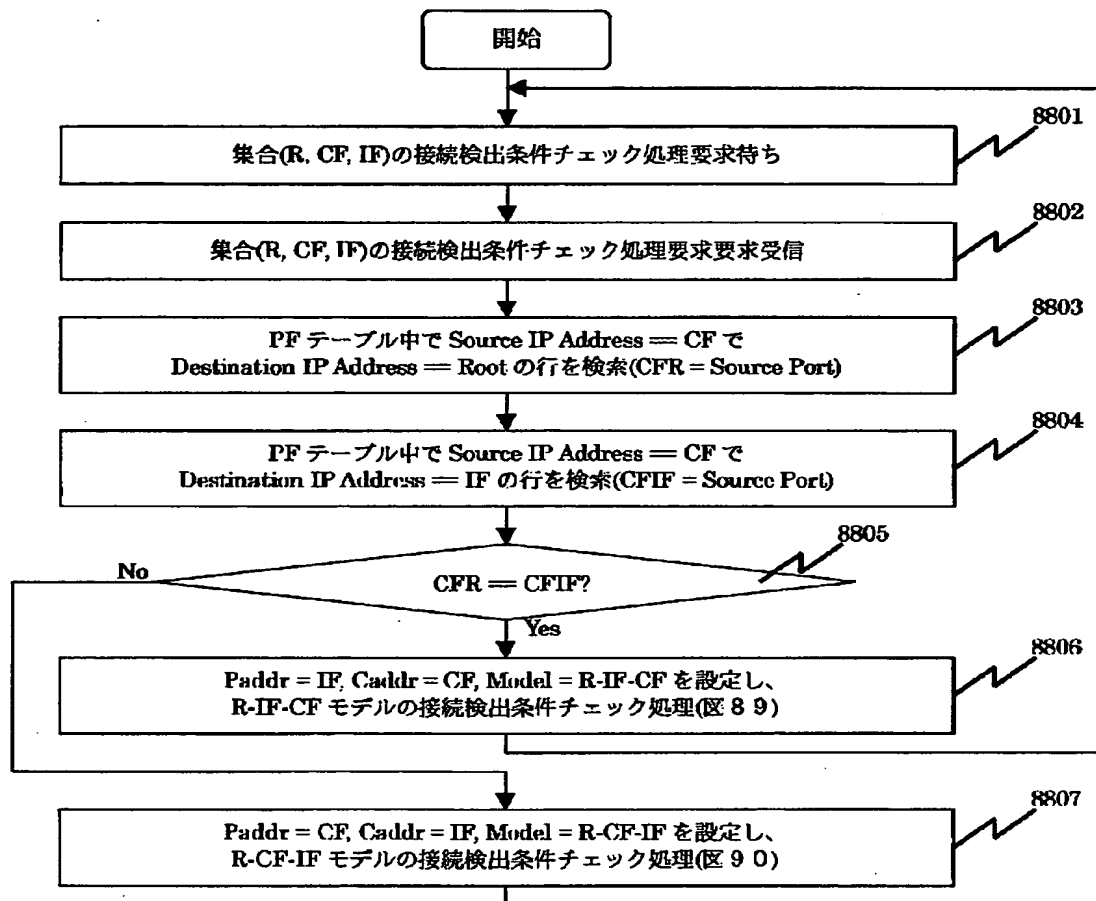
オートディスカバリモジュールの動作フローチャート 1 9
(TS テーブル作成(集合(R, CF, CF)の接続検出条件チェック処理)(図 4 6))



【図 8 8】

図 8 8

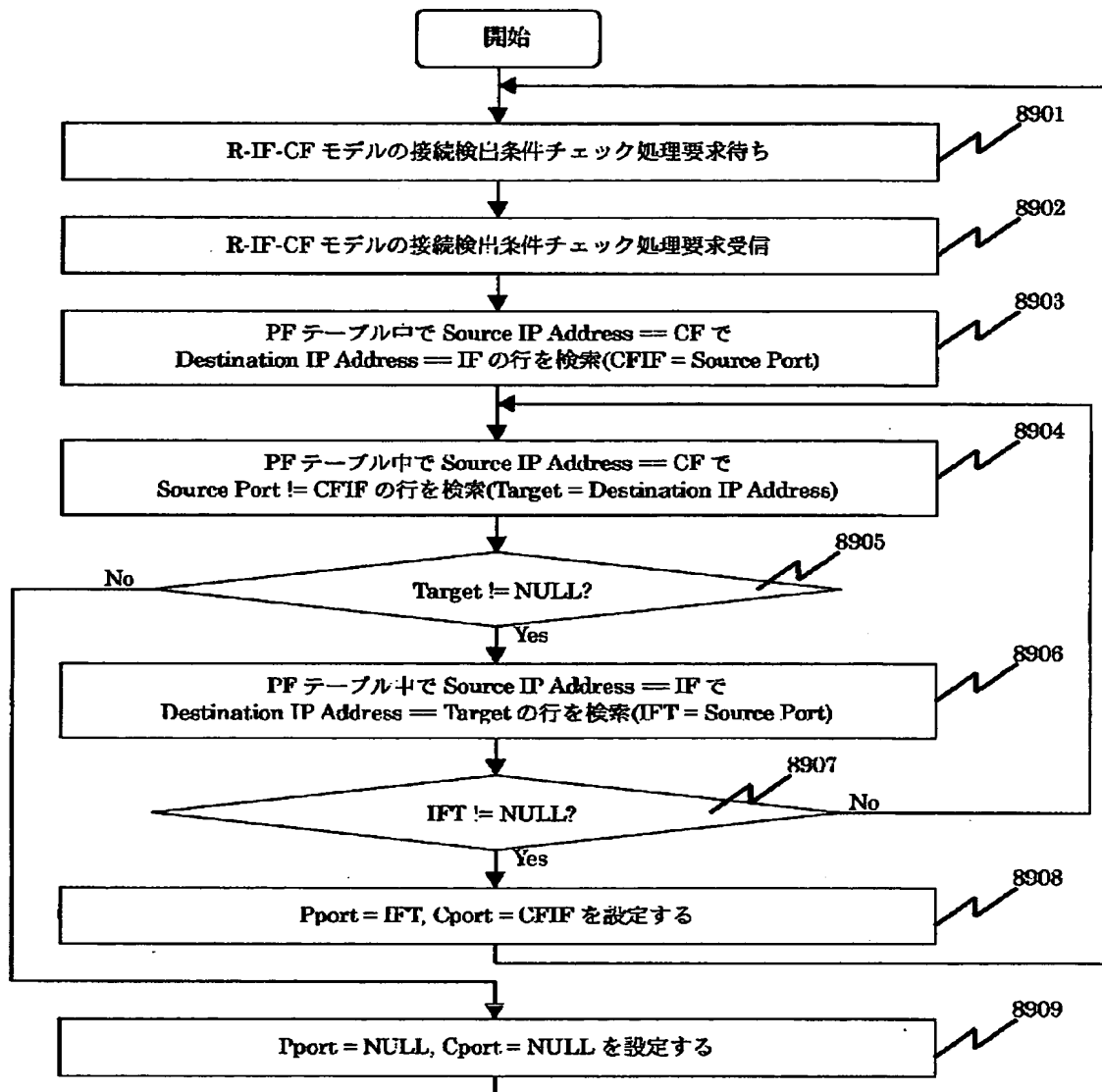
オートディスカバリモジュールの動作フローチャート 2 0
(TS テーブル作成(集合(R, CF, IF)の接続検出条件チェック処理)(図 4 6))



【図 8 9】

図 8 9

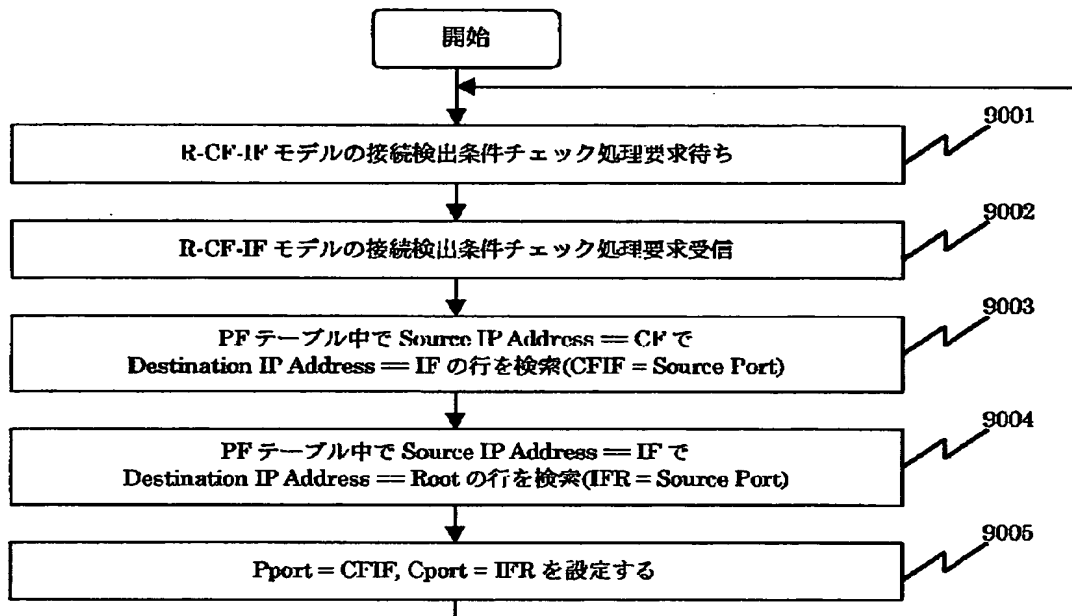
オートディスカバリモジュールの動作フローチャート 2 1
(TS テーブル作成(R-IF-CF モデルの接続検出条件チェック処理)(図 4 6))



【図 9 0】

図 9 0

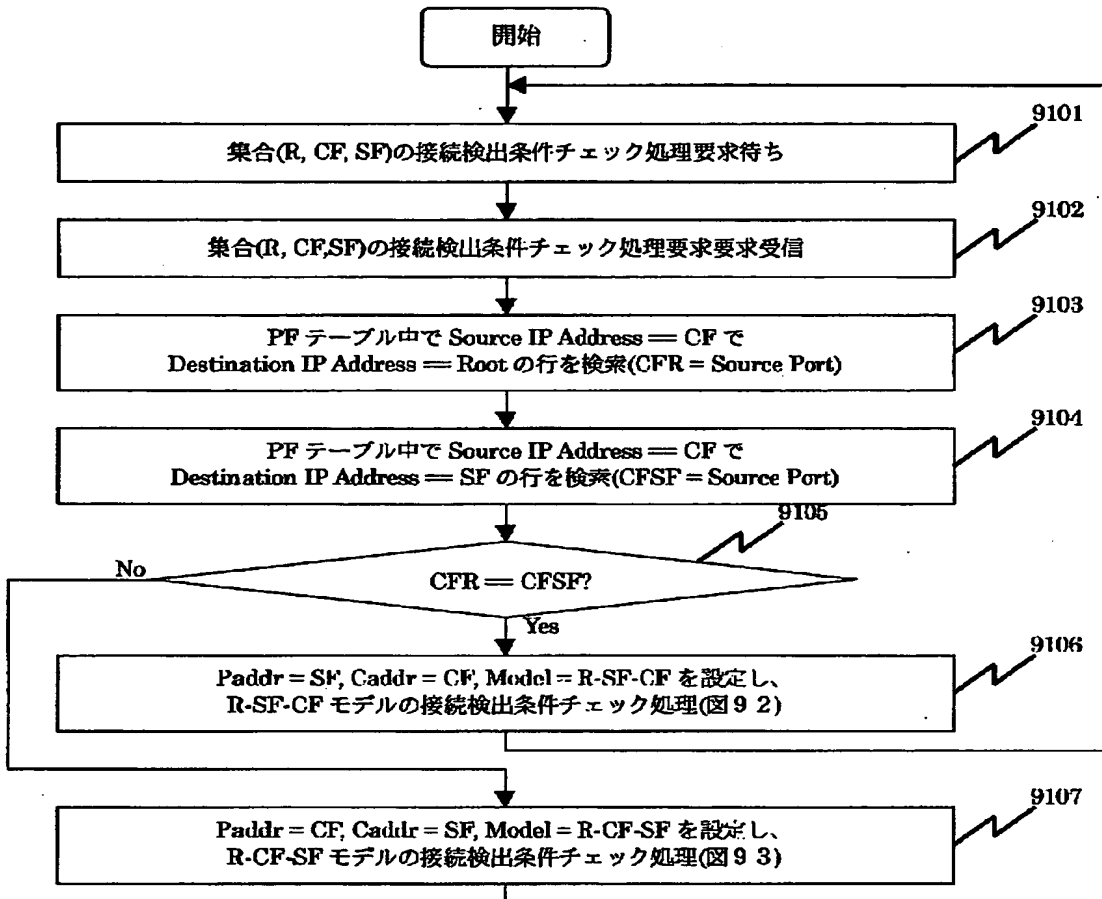
オートディスカバリモジュールの動作フローチャート 2 2
(TS テーブル作成(R-CF-IF モデルの接続検出条件チェック処理)(図 4 6))



【図 9 1】

図 9 1

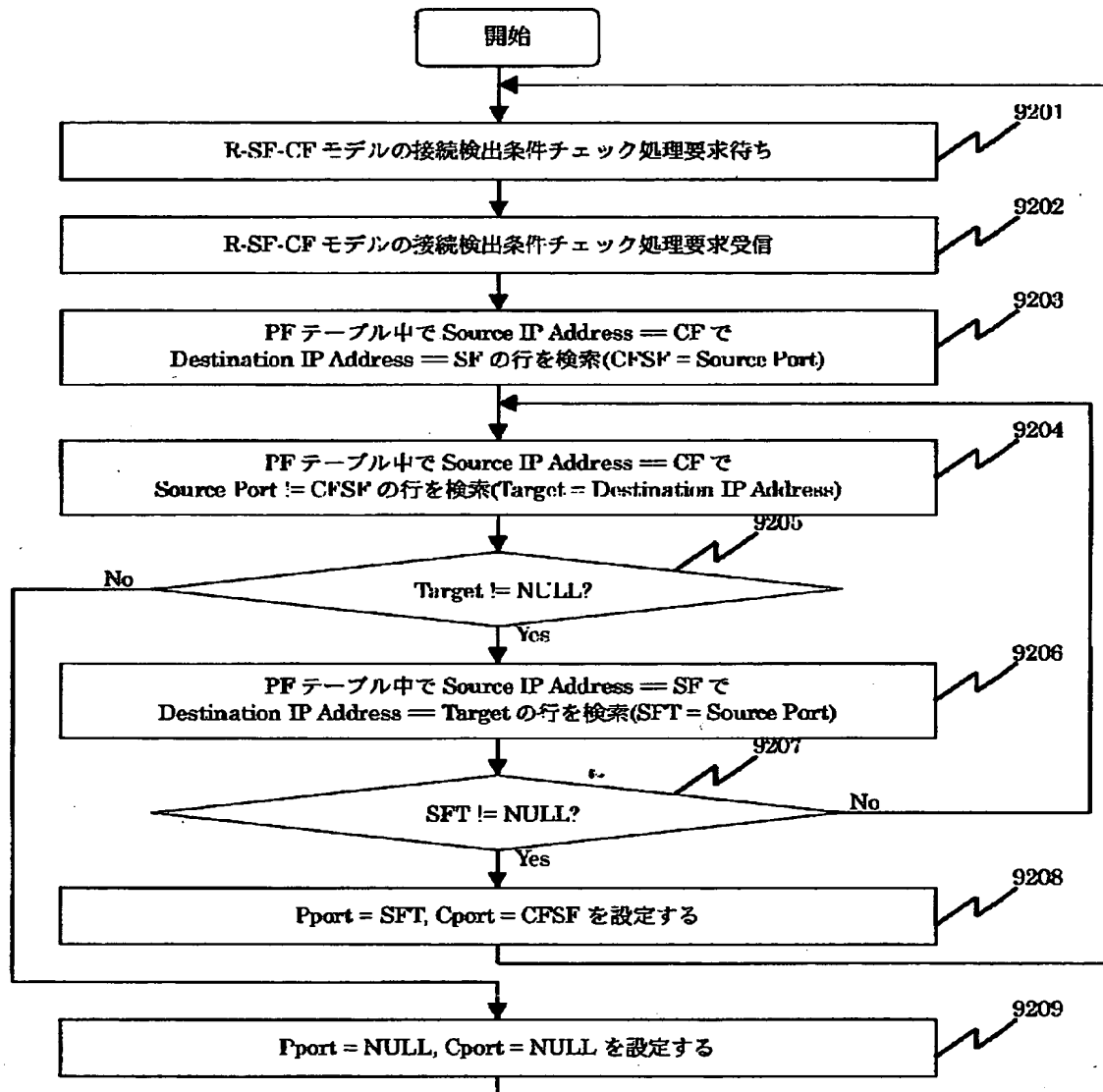
オートディスカバリモジュールの動作フローチャート 2 3
(TS テーブル作成(集合(R, CF, SF)の接続検出条件チェック処理)(図 4 6))



【図 9 2】

図 9 2

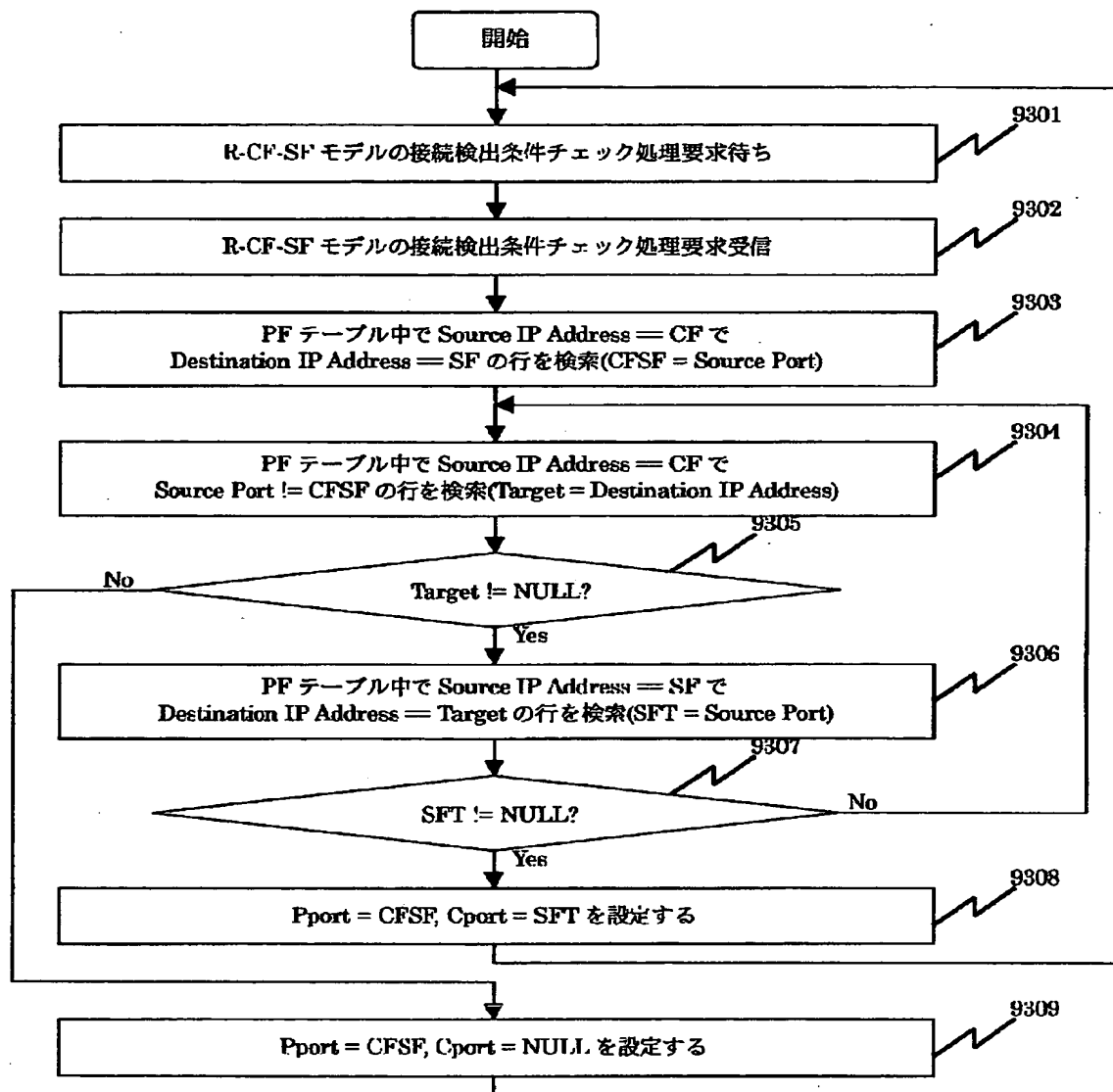
オートディスカバリモジュールの動作フローチャート 2 4
 (TS テーブル作成(R-SF-CF モデルの接続検出条件チェック処理)(図 4 6))



【図 9 3】

図 9 3

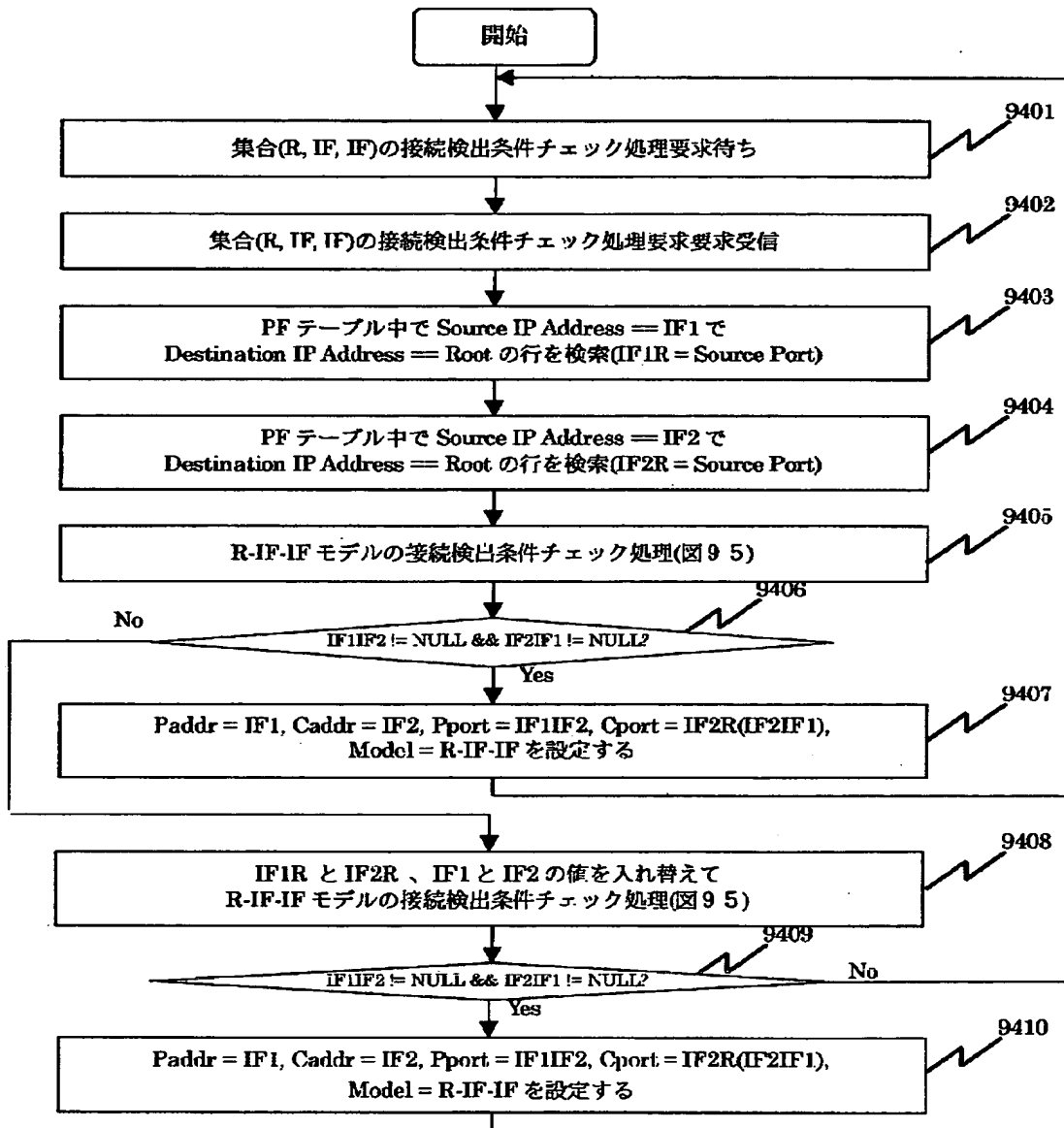
オートディスカバリモジュールの動作フローチャート 2 5
 (TS テーブル作成(R-CF-SF モデルの接続検出条件チェック処理)(図 4 6))



【図 9 4】

図 9 4

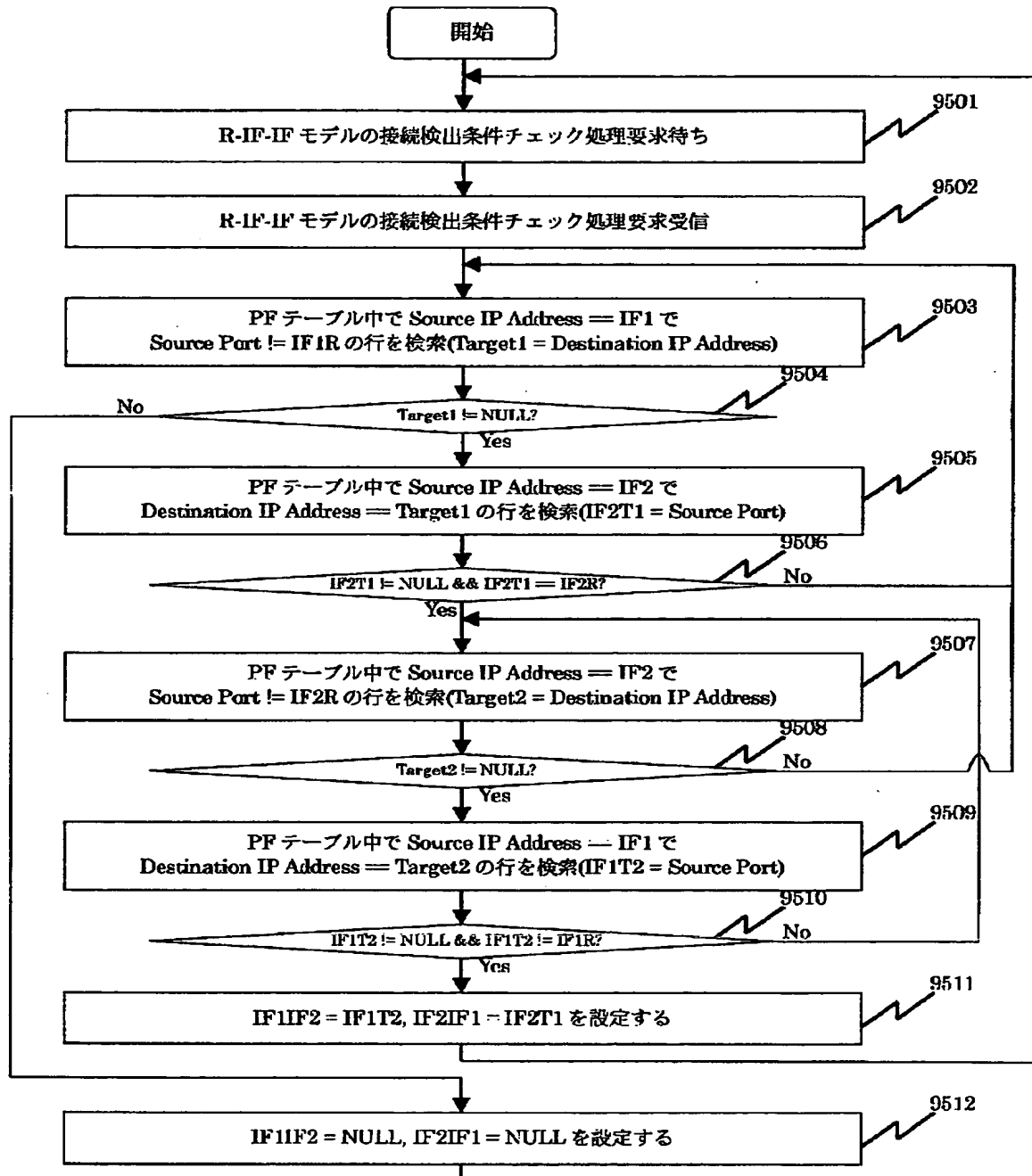
オートディスカバリモジュールの動作フローチャート 26
(TS テーブル作成(集合(R, IF, IF)の接続検出条件チェック処理)(図 4 6))



【図 9 5】

図 9 5

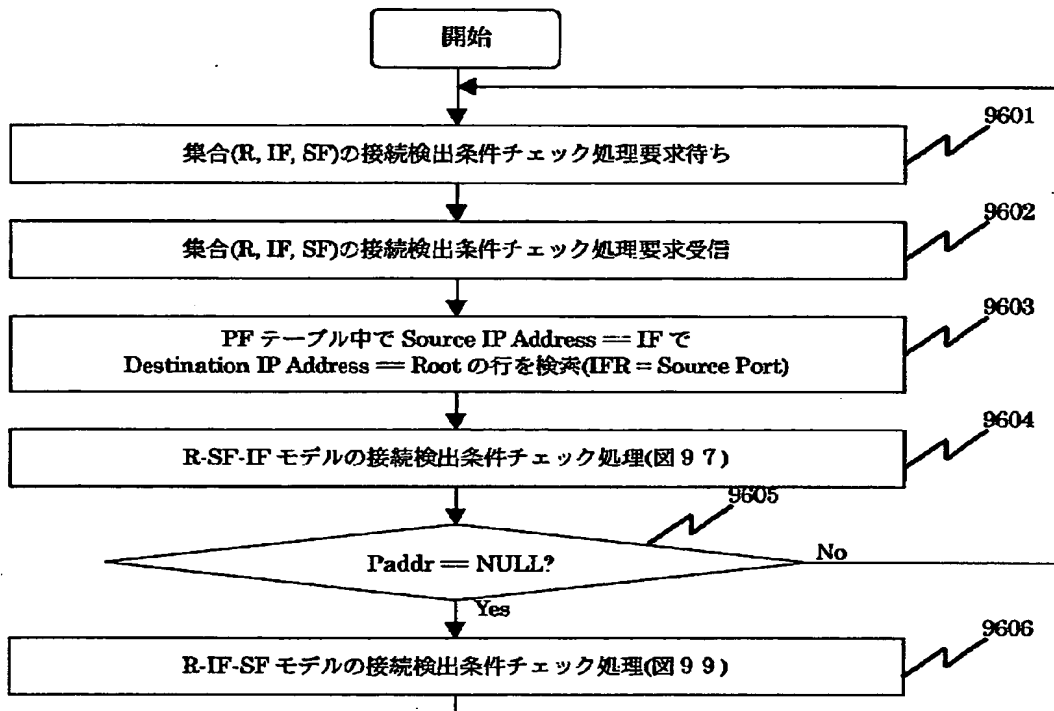
オートディスカバリモジュールの動作フローチャート 27
(TS テーブル作成(R-IF-IF モデルの接続検出条件チェック処理)(図 1 6))



【図 9 6】

図 9 6

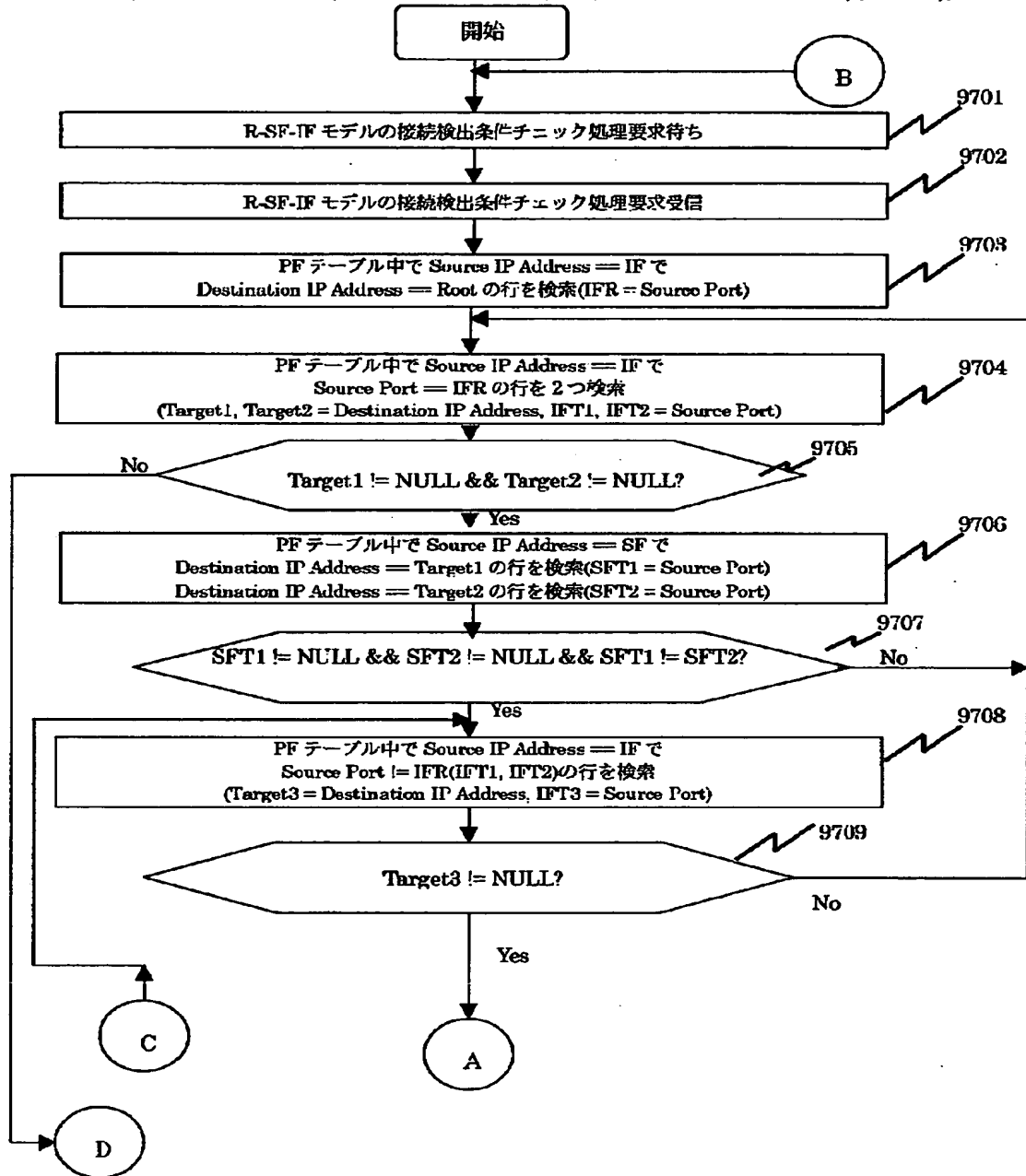
オートディスカバリモジュールの動作フローチャート 2 8
(TS テーブル作成(集合(R, IF, SF)の接続検出条件チェック処理)(図 4 6))



【図 9 7】

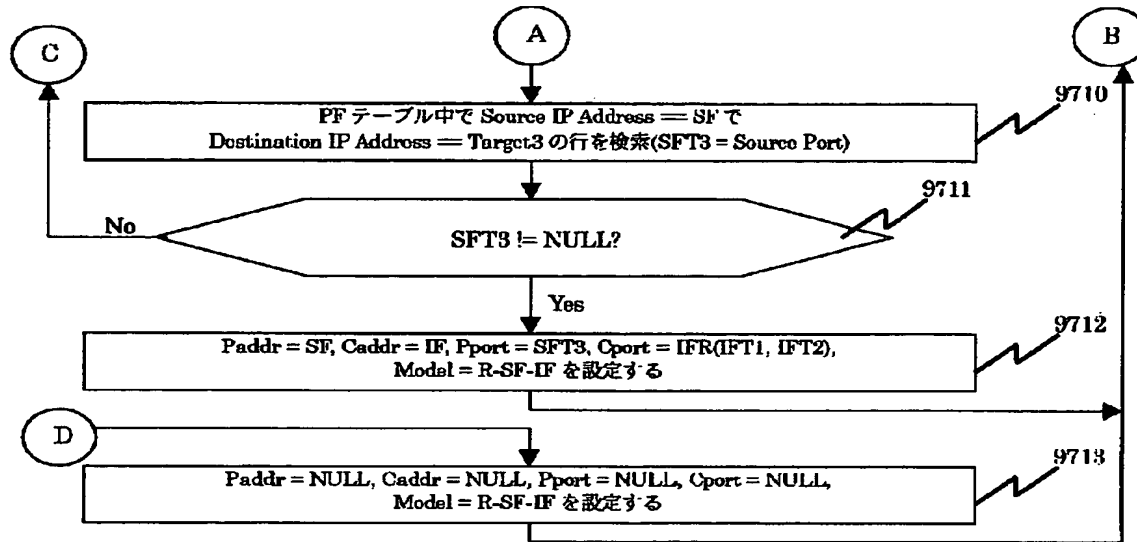
図 9 7

オートディスクバリモジュールの動作フローチャート 29
(TS テーブル作成(R-SF-IF モデルの接続検出条件チェック処理)(図 4 6))



【図 98】

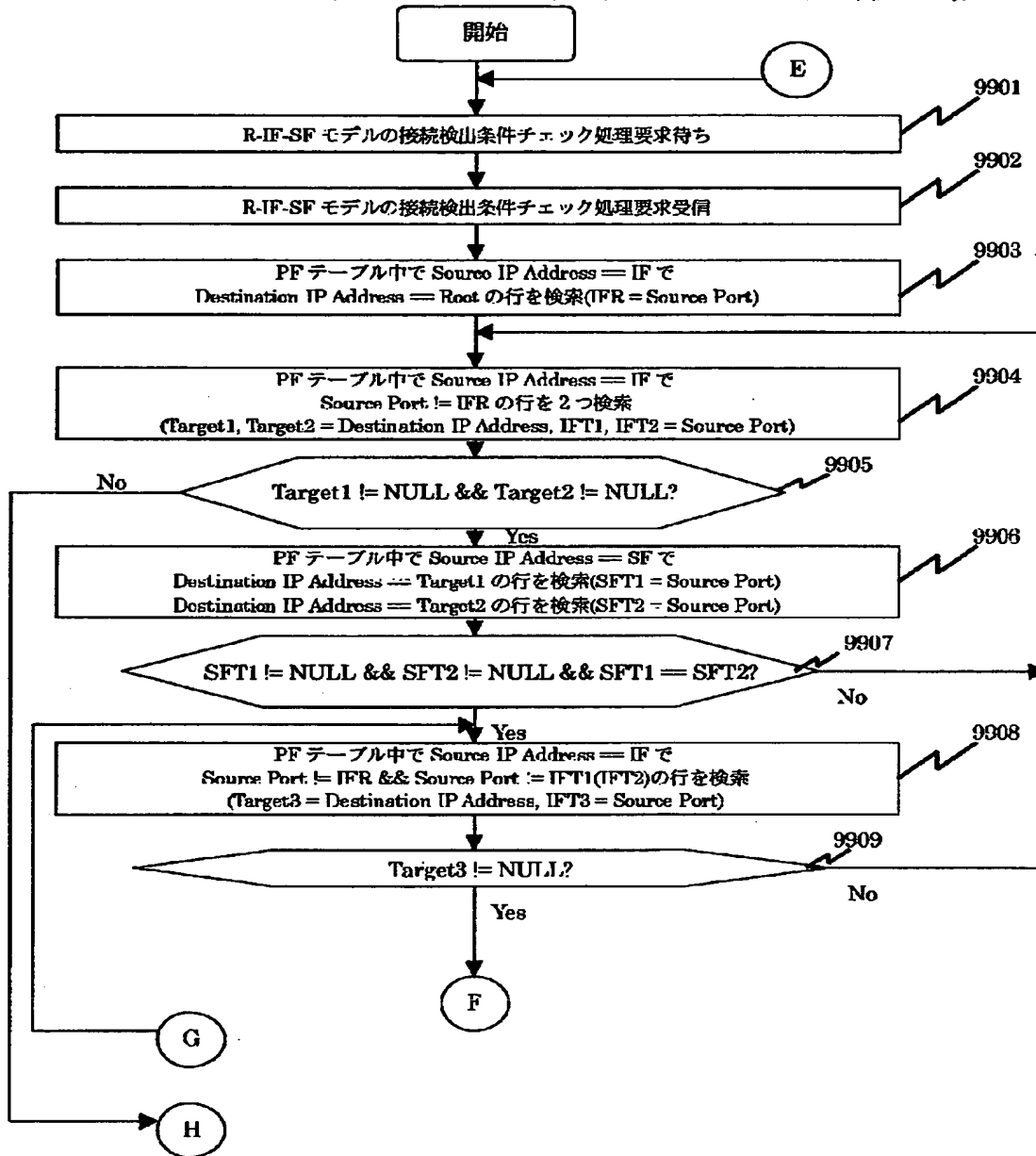
図 98



【図 99】

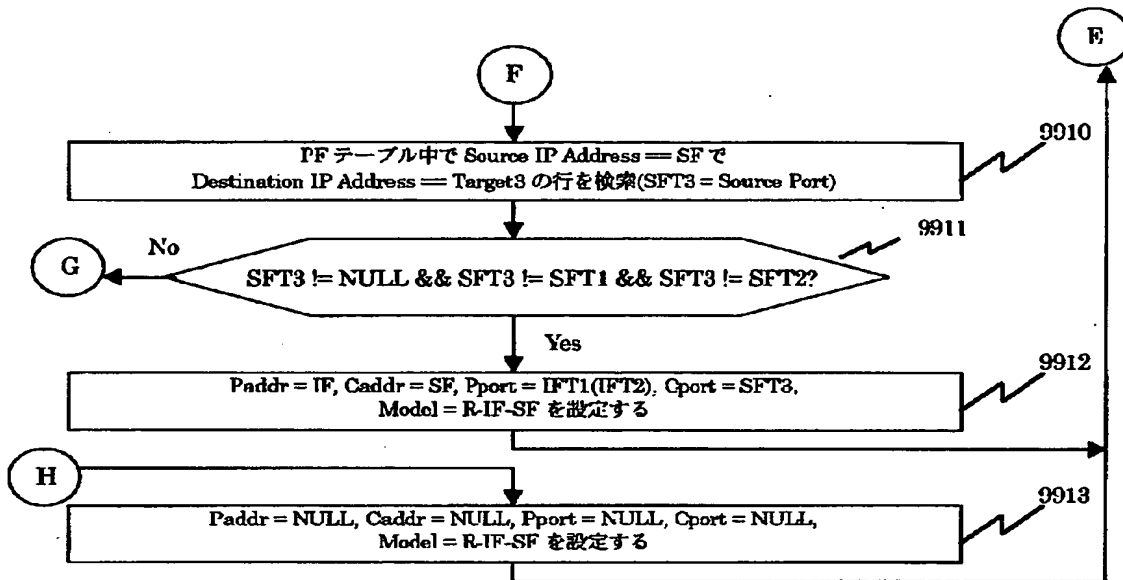
図 99

オートディスカバリモジュールの動作フローチャート 30
(TS テーブル作成(R-IF-SF モデルの接続検出条件チェック処理)(図 46))



【図 1 0 0】

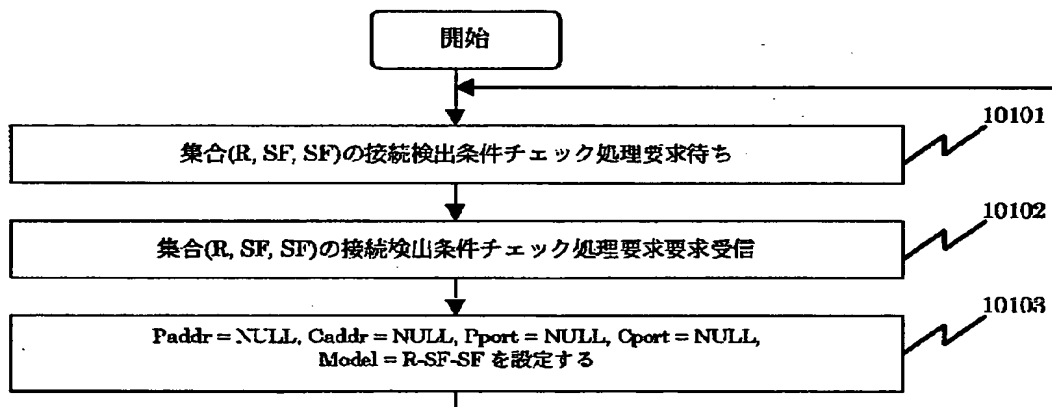
図 1 0 0



【図 1 0 1】

図 1 0 1

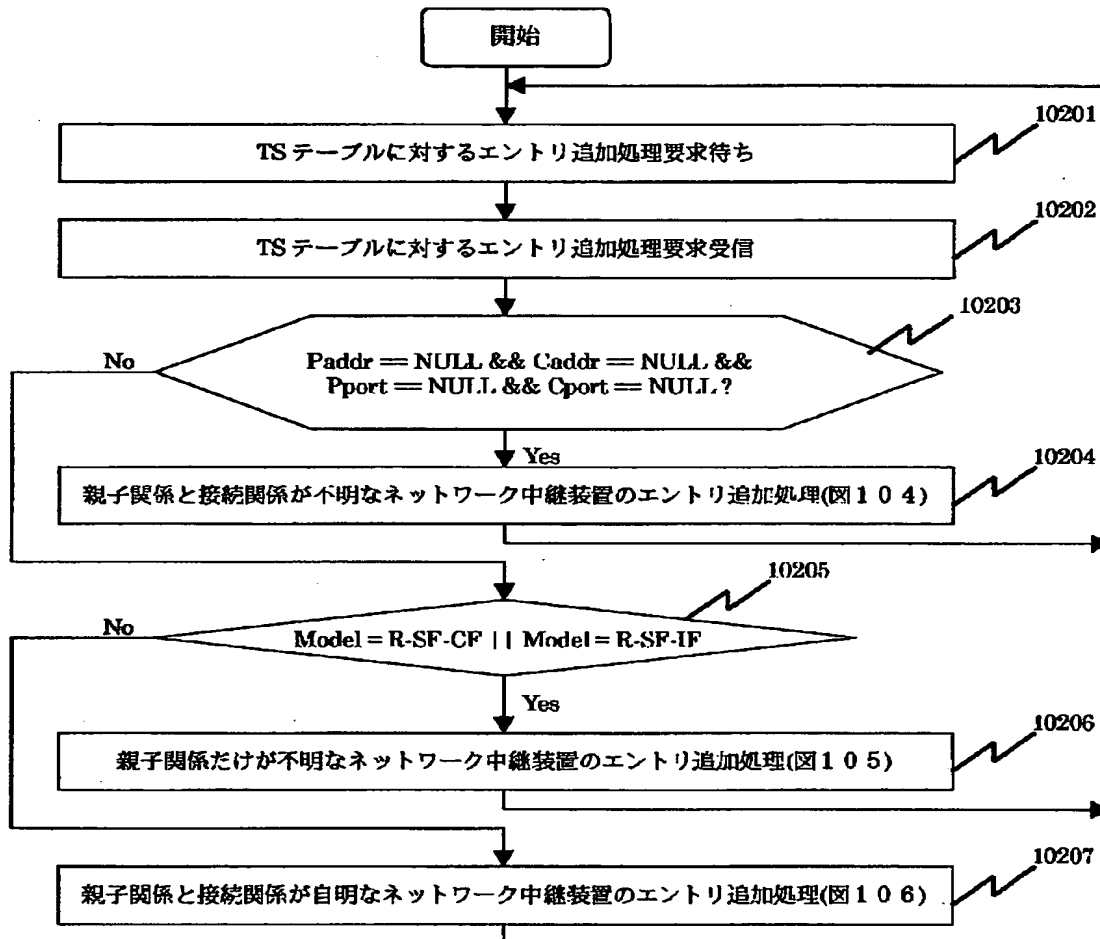
オートディスカバリモジュールの動作フローチャート 3 1
(TS テーブル作成(集合(R, SF, SF)の接続検出条件チェック処理)(図 4 6))



【図 1 0 2】

図 1 0 2

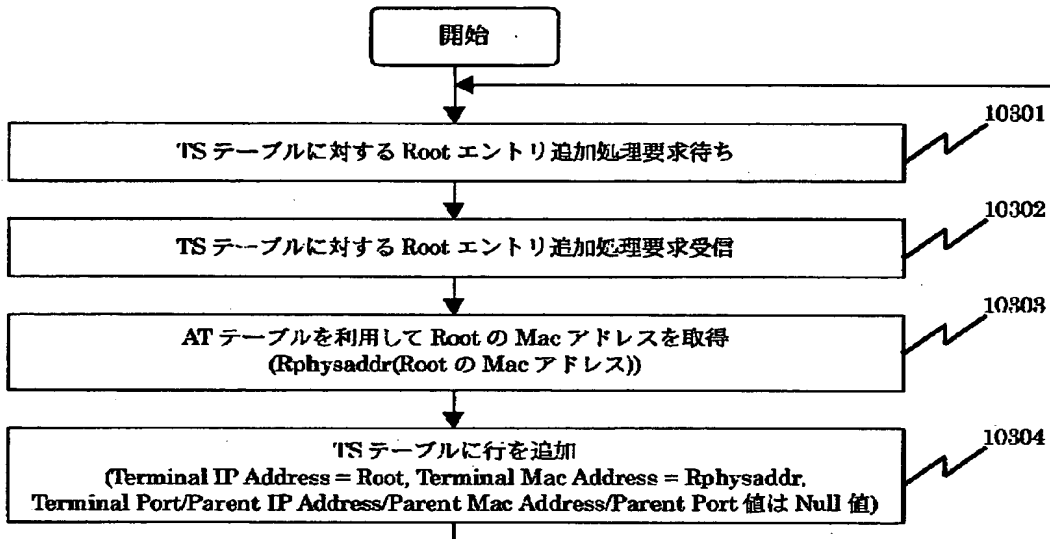
オートディスカバリモジュールの動作フローチャート 3 2
(TS テーブル作成(TS テーブルに対するエントリ追加処理))



【図 1 0 3】

図 1 0 3

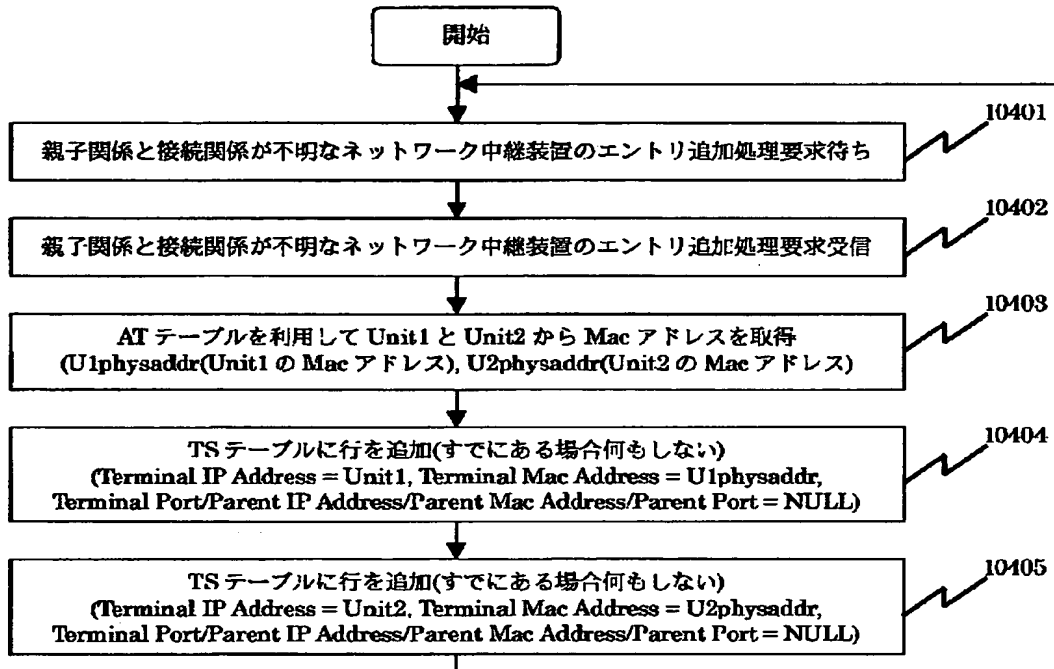
オートディスカバリモジュールの動作フローチャート 3 3
(TS テーブル作成(TS テーブルに対する Root エントリ追加処理))



【図 1 0 4】

図 1 0 4

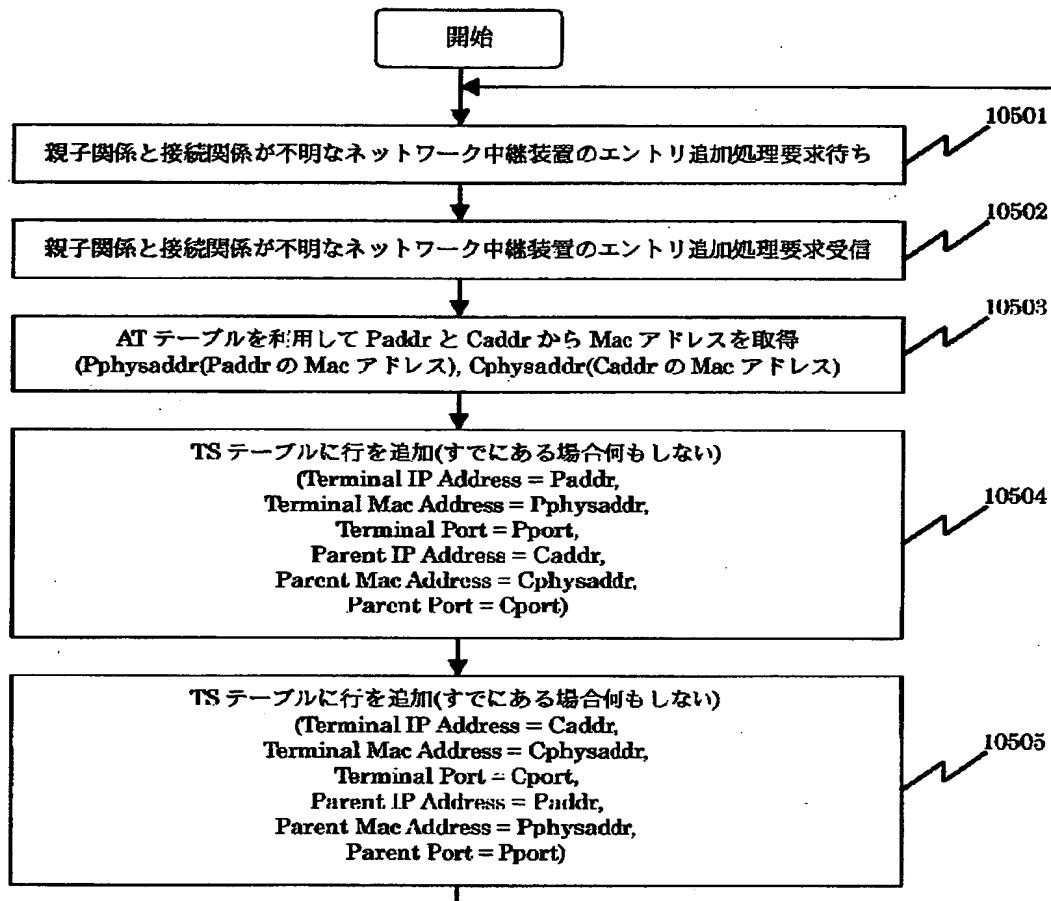
オートディスカバリモジュールの動作フローチャート 3 4
(TS テーブル作成(親子関係と接続関係が不明なネットワーク中継装置のエントリ追加処理))



【図 105】

図 105

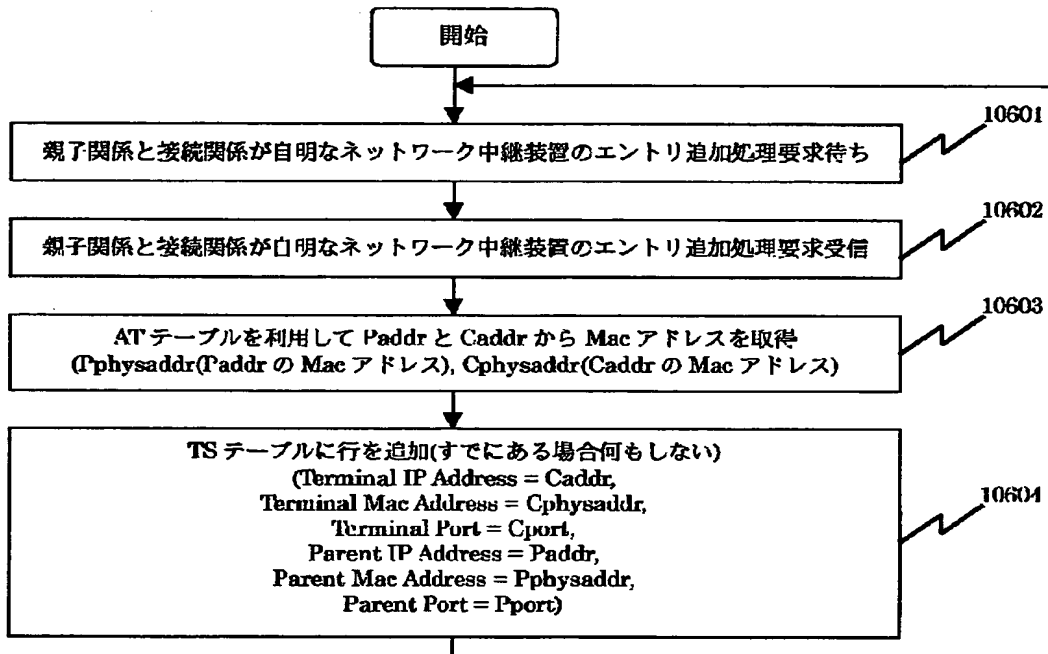
オートディスカバリモジュールの動作フローチャート 35
(TS テーブル作成(親子関係だけが不明なネットワーク中継装置のエントリ追加処理))



【図 1 0 6】

図 1 0 6

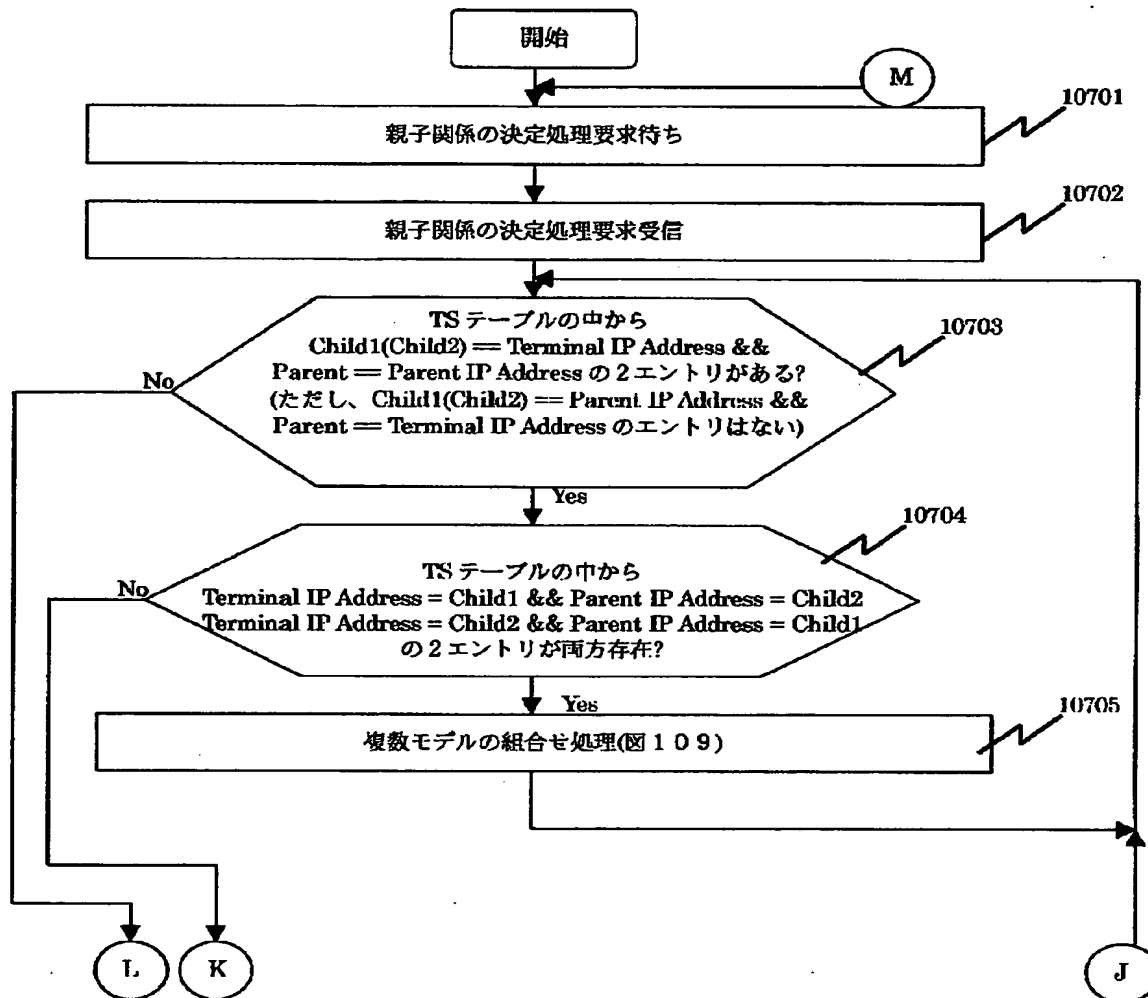
オートディスカバリモジュールの動作フローチャート 3 6
(TS テーブル作成(親子関係と接続関係が自明なネットワーク中継装置のエントリ追加処理))



【図 1 0 7】

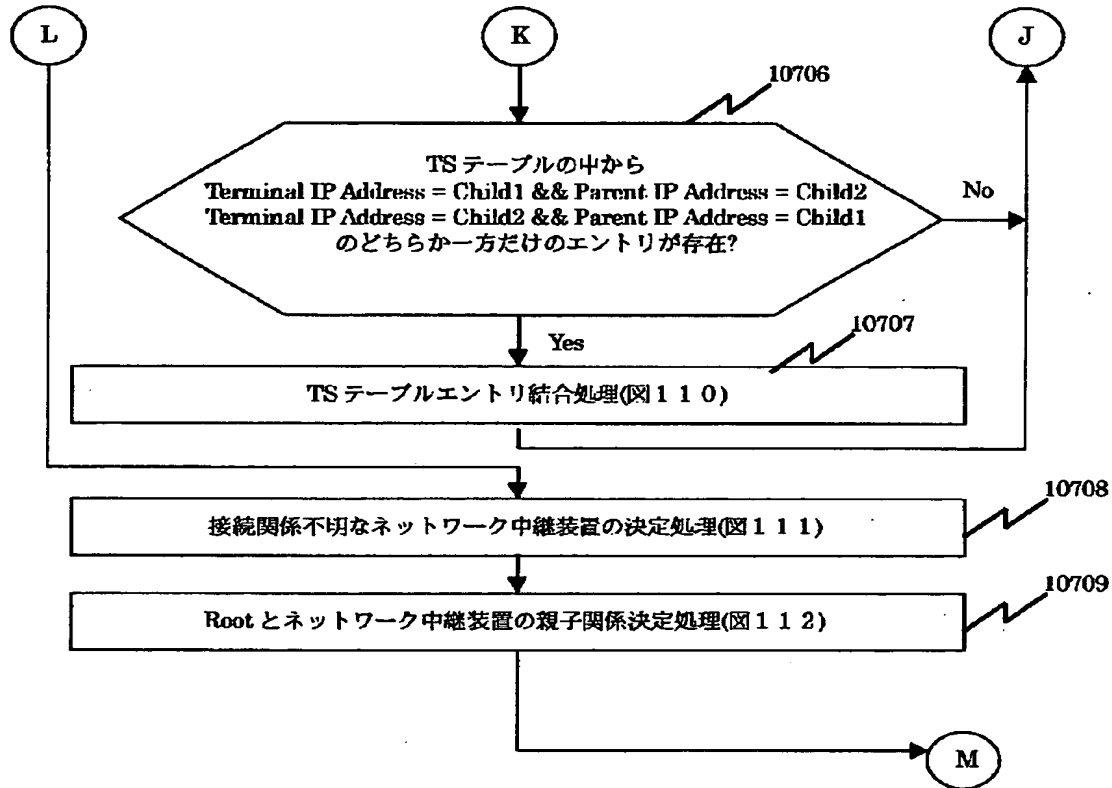
図 1 0 7

オートディスカバリモジュールの動作フローチャート 3 7
(TS テーブル作成(親子関係の決定処理))



【図 1 0 8】

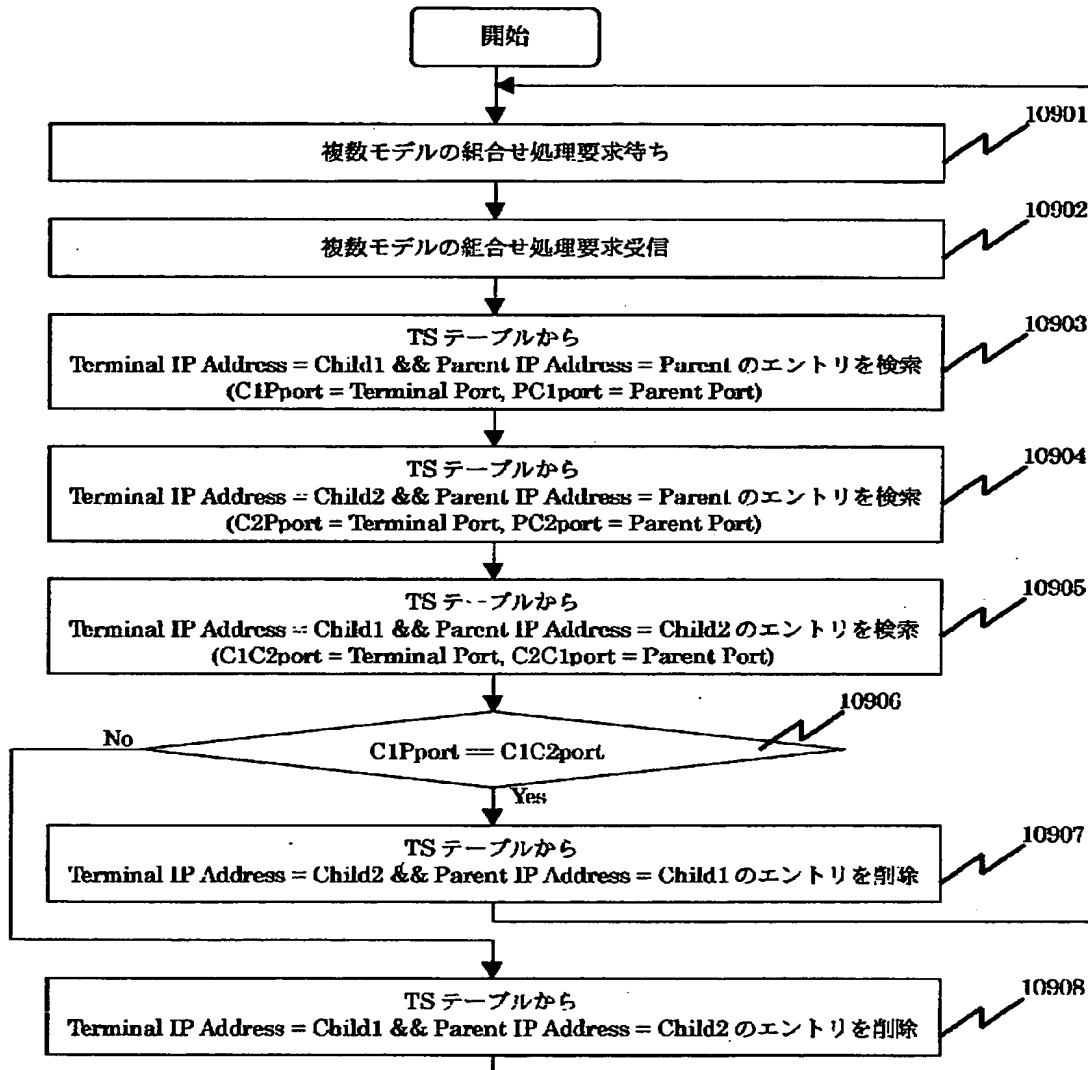
図 1 0 8



【図 1 0 9】

図 1 0 9

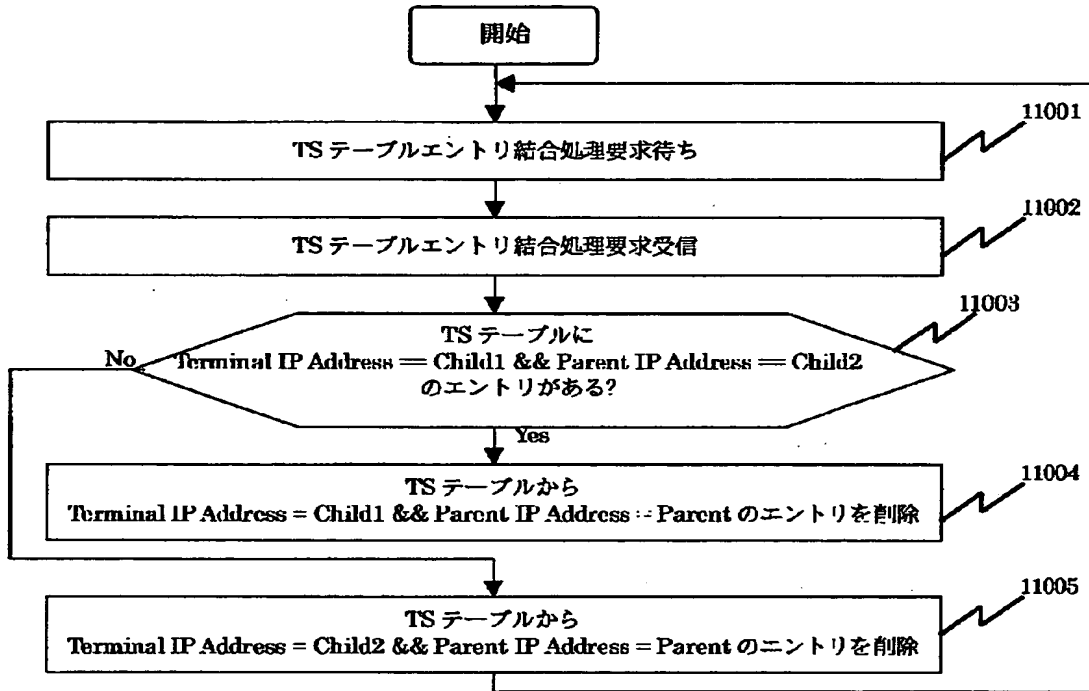
オートディスカバリモジュールの動作フローチャート 3 8
(TS テーブル作成(複数モデルの組合せ処理(図 5 5)))



【図 1 1 0】

図 1 1 0

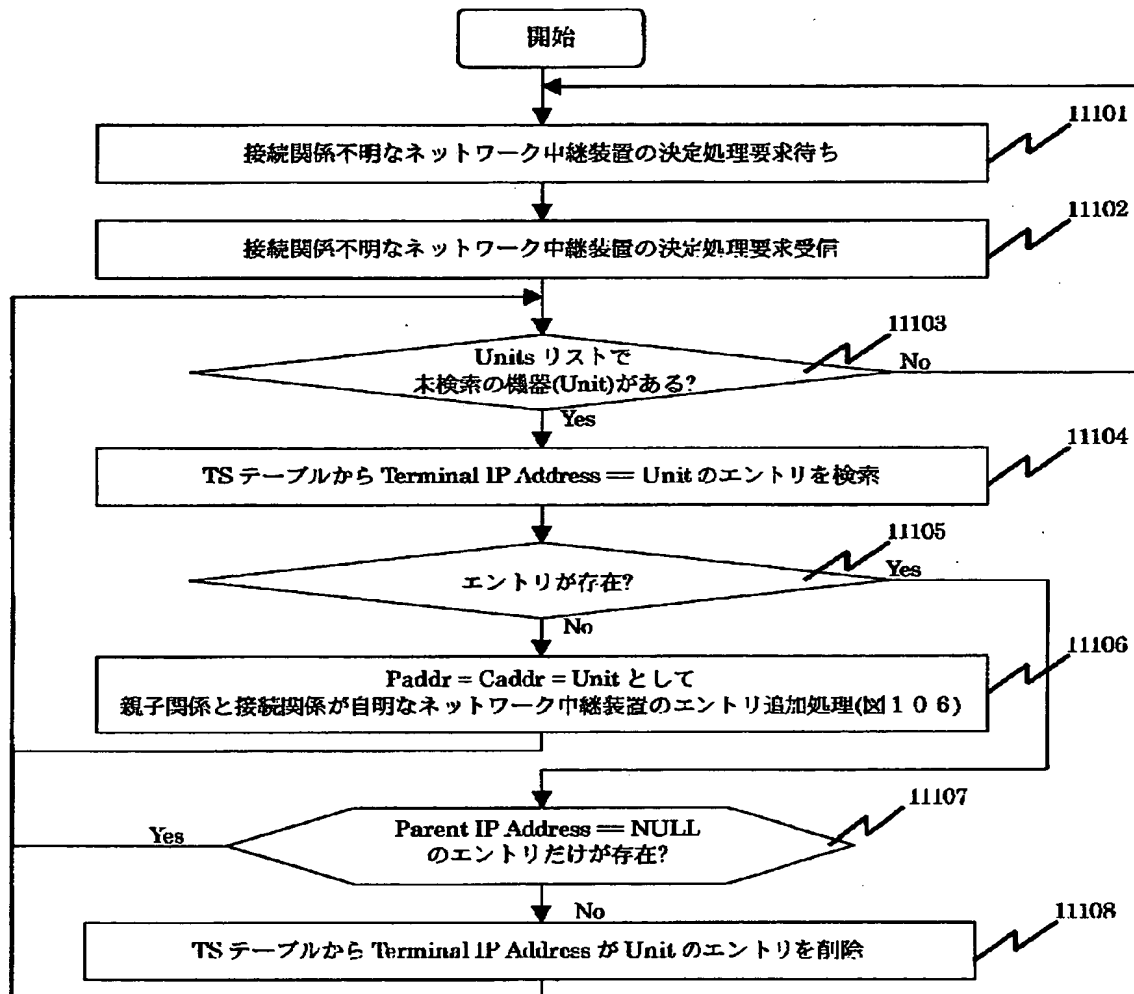
オートディスカバリモジュールの動作フローチャート 3 9
(TS テーブル作成(TS テーブルエントリ結合処理))



【図 1 1 1】

図 1 1 1

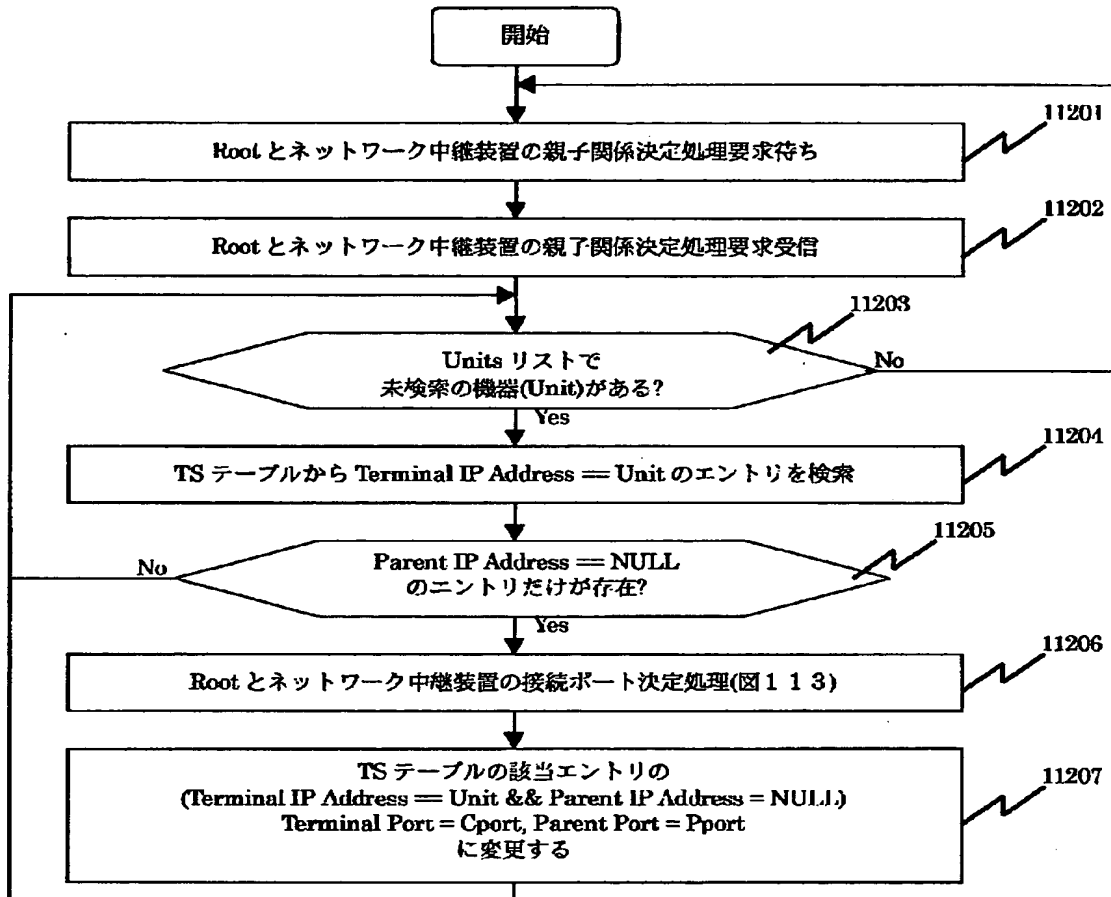
オートディスカバリモジュールの動作フローチャート 40
(TS テーブル作成(接続関係不明なネットワーク中継装置の決定処理))



【図 1 1 2】

図 1 1 2

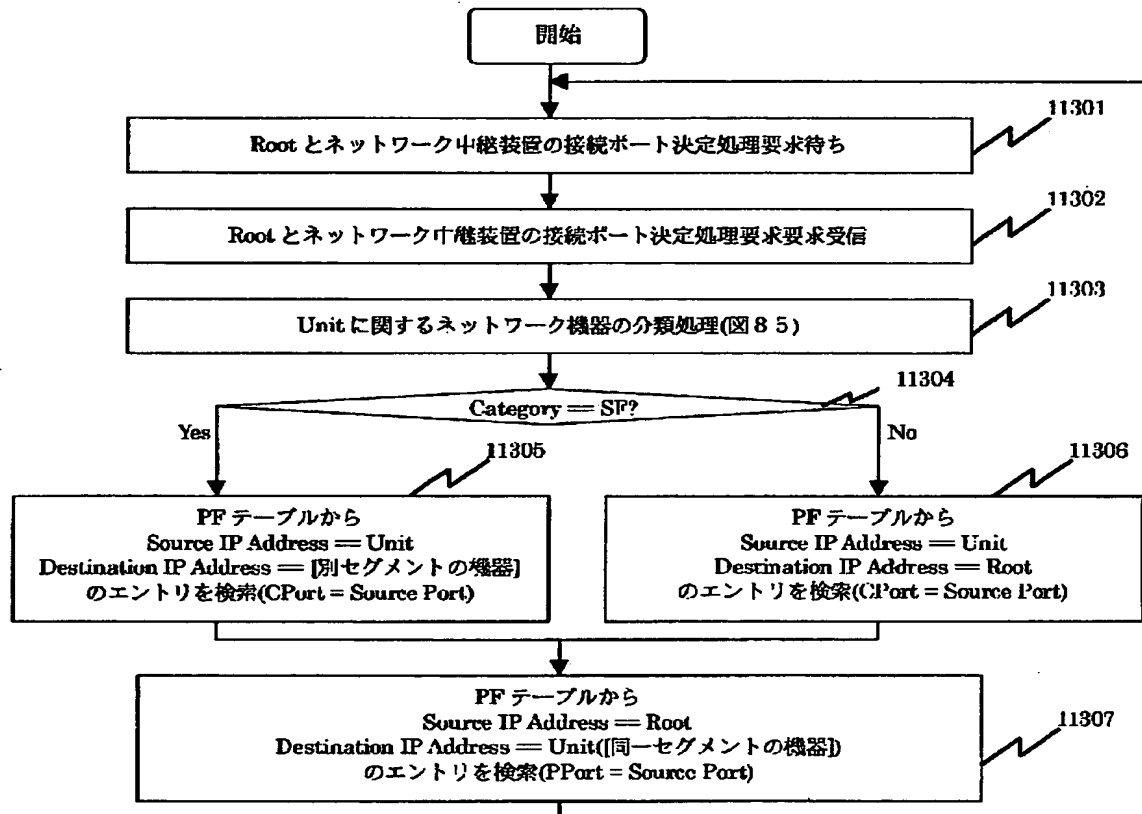
オートディスカバリモジュールの動作フローチャート 4 1
(TS テーブル作成(Root とネットワーク中継装置の親子関係決定処理))



【図 1 1 3】

図 1 1 3

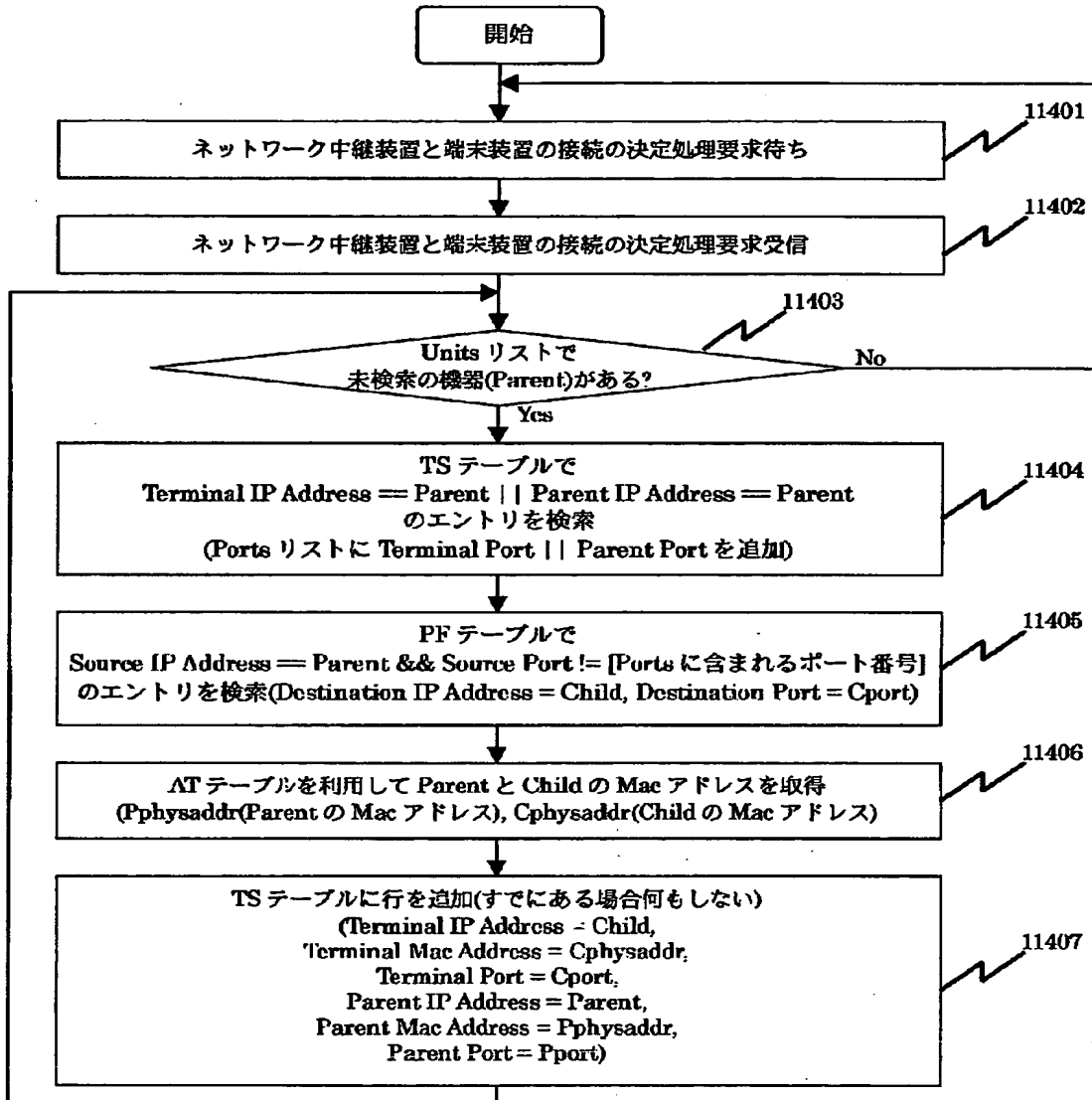
オートディスカバリモジュールの動作フローチャート 4 2
(TS テーブル作成(Root とネットワーク中継装置の接続ポート決定処理))



【図 1 1 4】

図 1 1 4

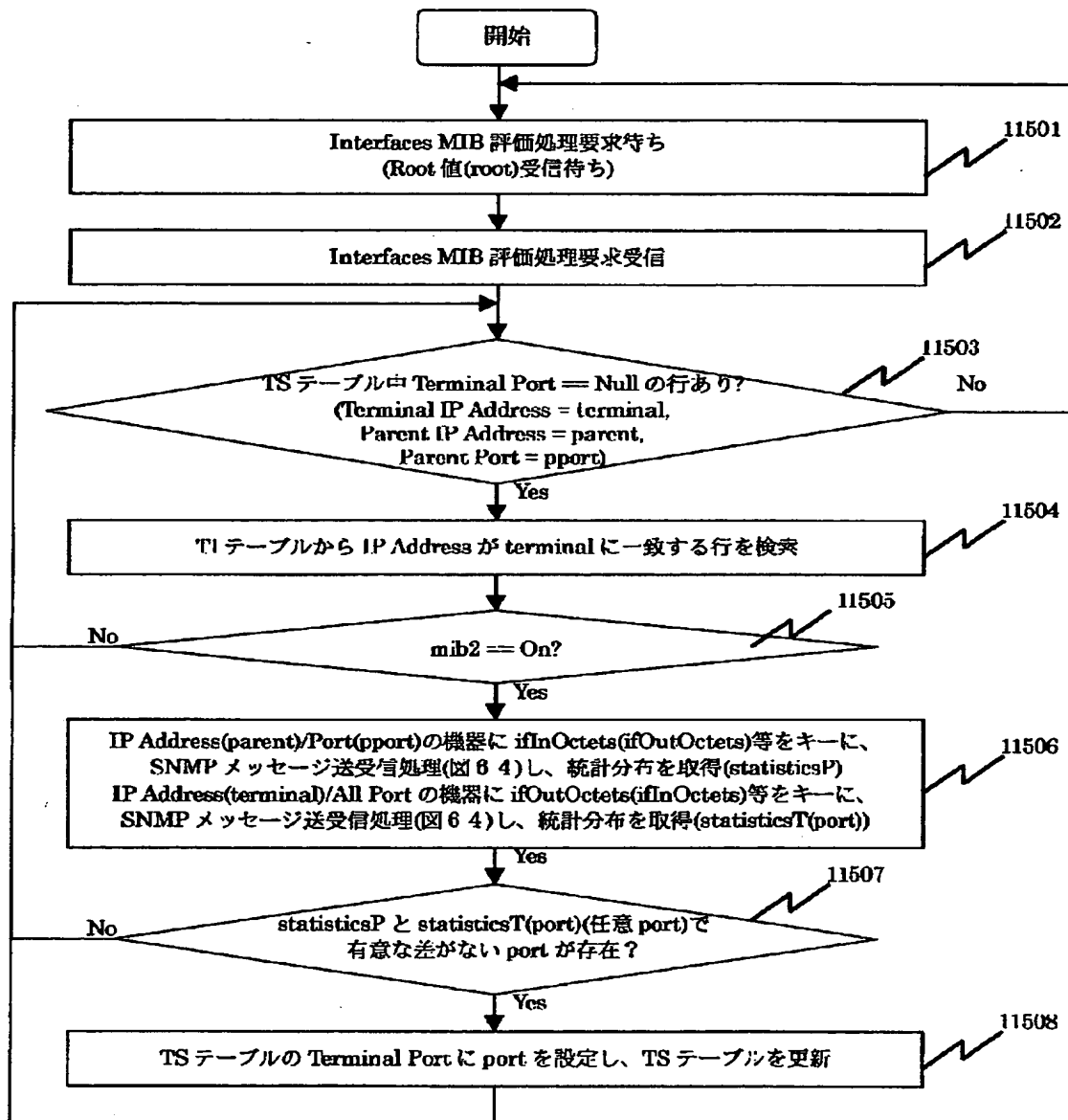
オートディスカバリモジュールの動作フローチャート 4 3
(TS テーブル作成(ネットワーク中継装置と端末装置の接続の決定処理))



【図 1 1 5】

図 1 1 5

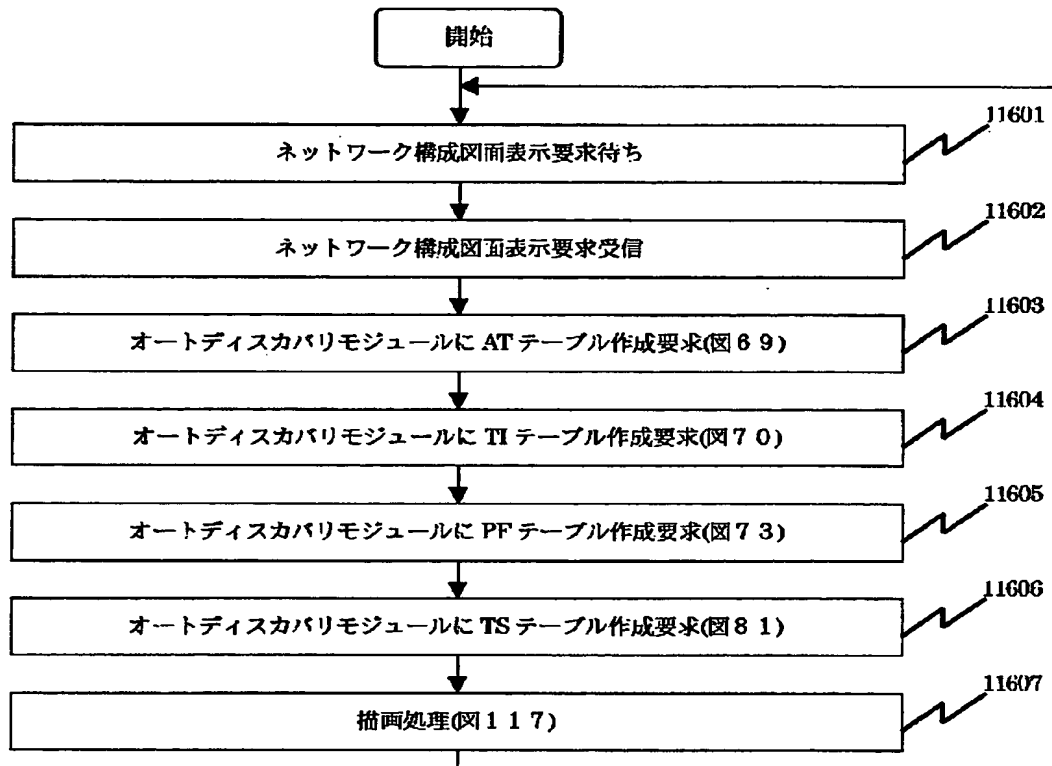
オートディスカバリモジュールの動作フローチャート 4 4
(TS テーブル作成(Interfaces MIB 評価処理))



【図 1 1 6】

図 1 1 6

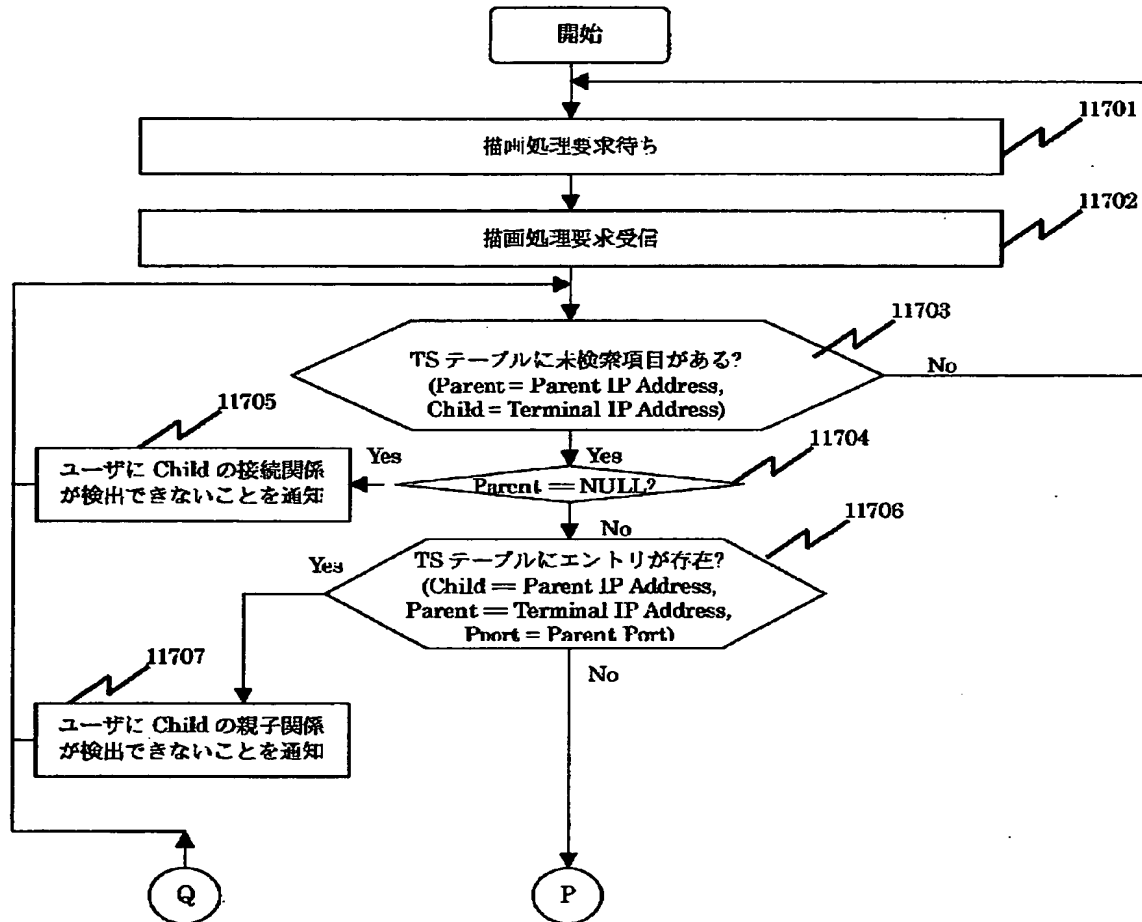
図面表示プログラムの動作フローチャート 1
(ネットワーク構成図面表示処理)



【図 1 1 7】

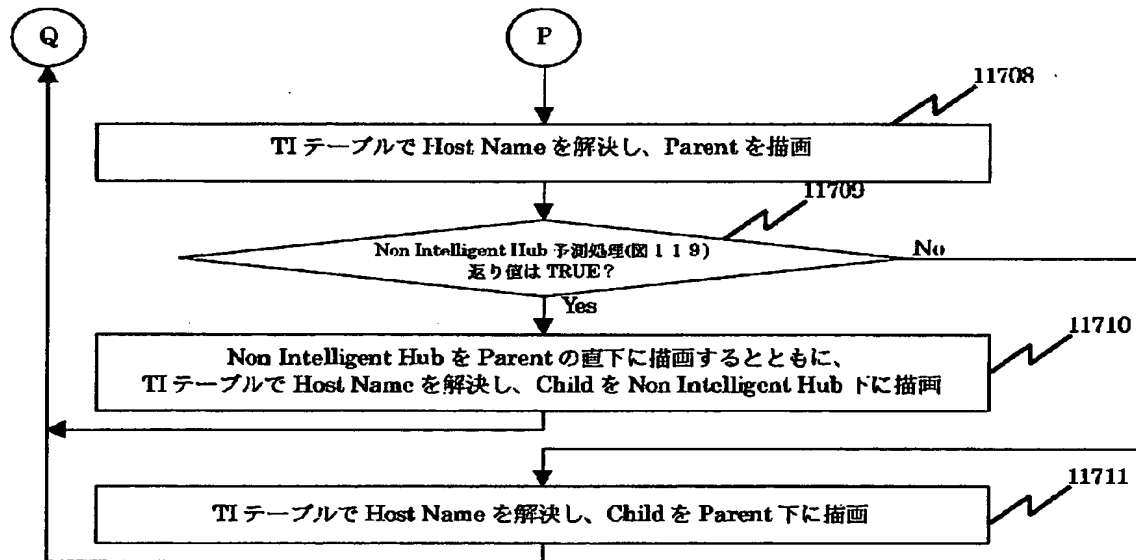
図 1 1 7

図面表示プログラムの動作フローチャート 2
(ネットワーク構成図面表示(描画処理))



【図 118】

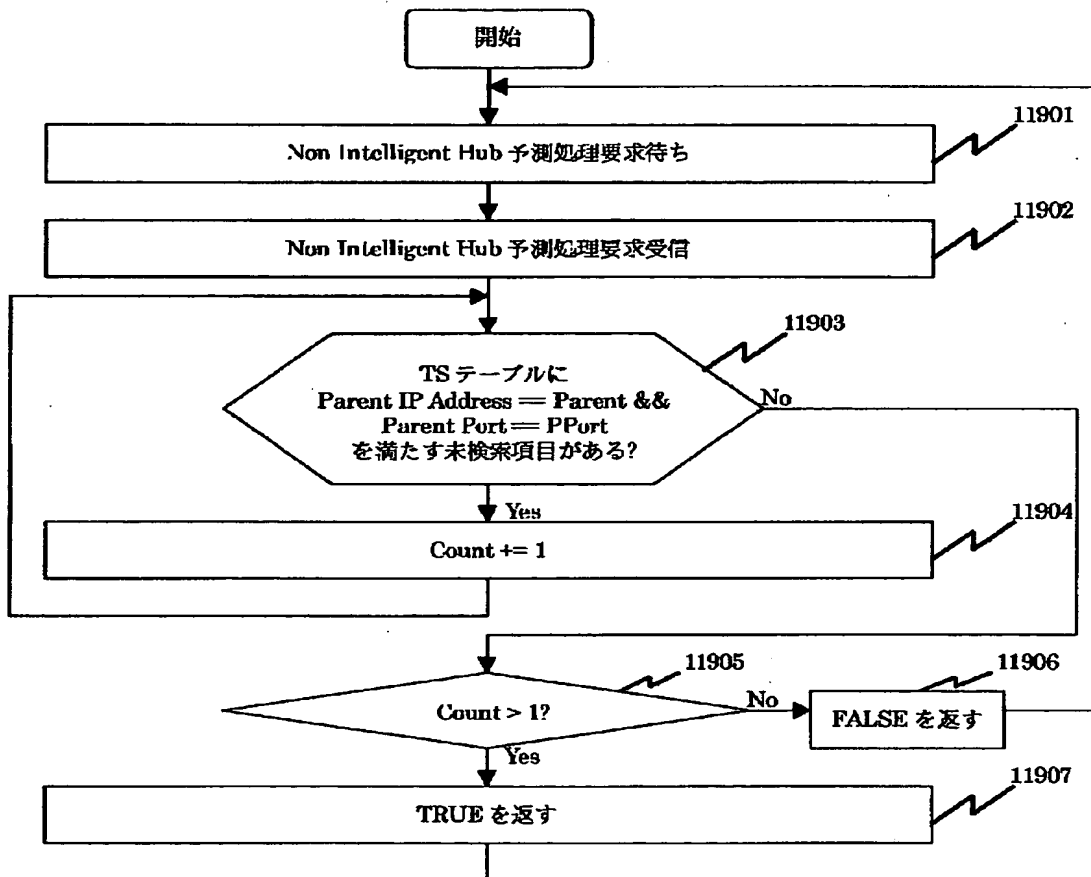
図 118



【図 1 1 9】

図 1 1 9

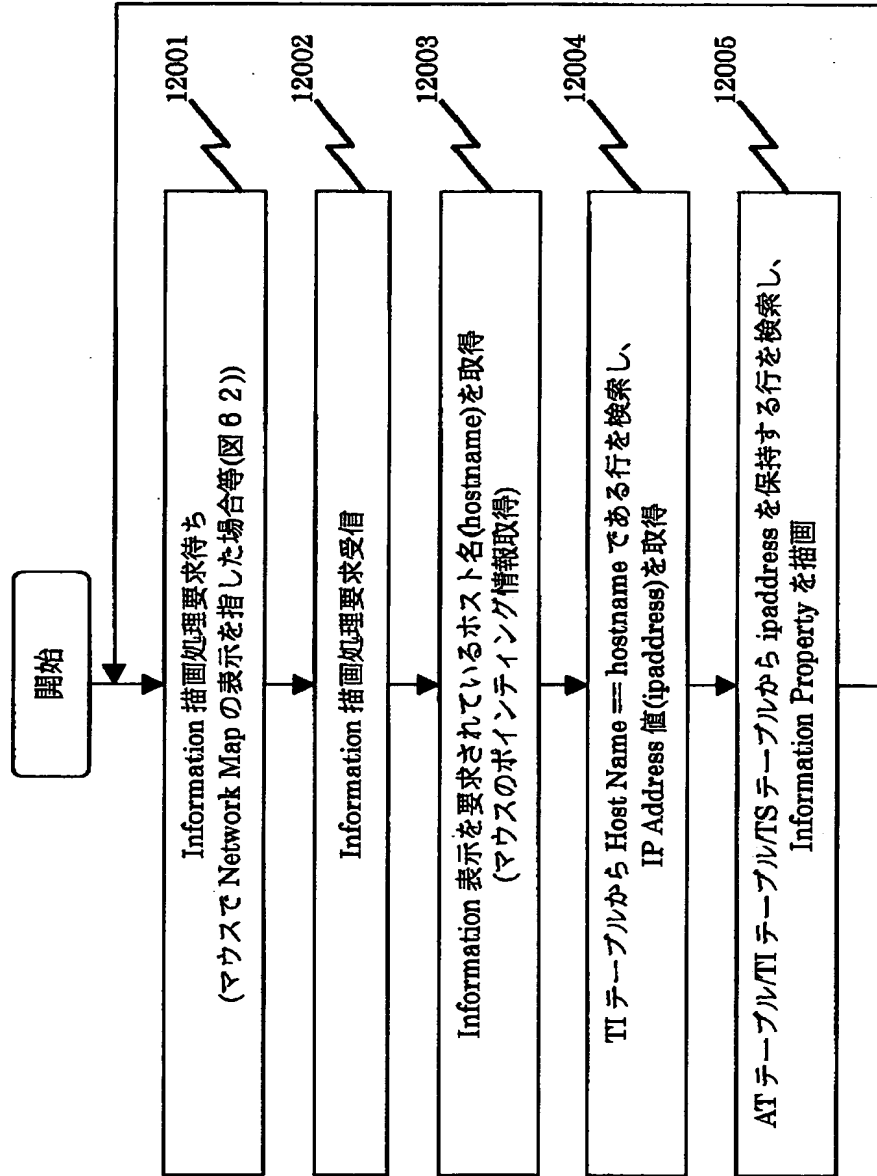
図面表示プログラムの動作フローチャート 3
(描画(Non Intelligent Hub 予測処理))



【図120】

図120

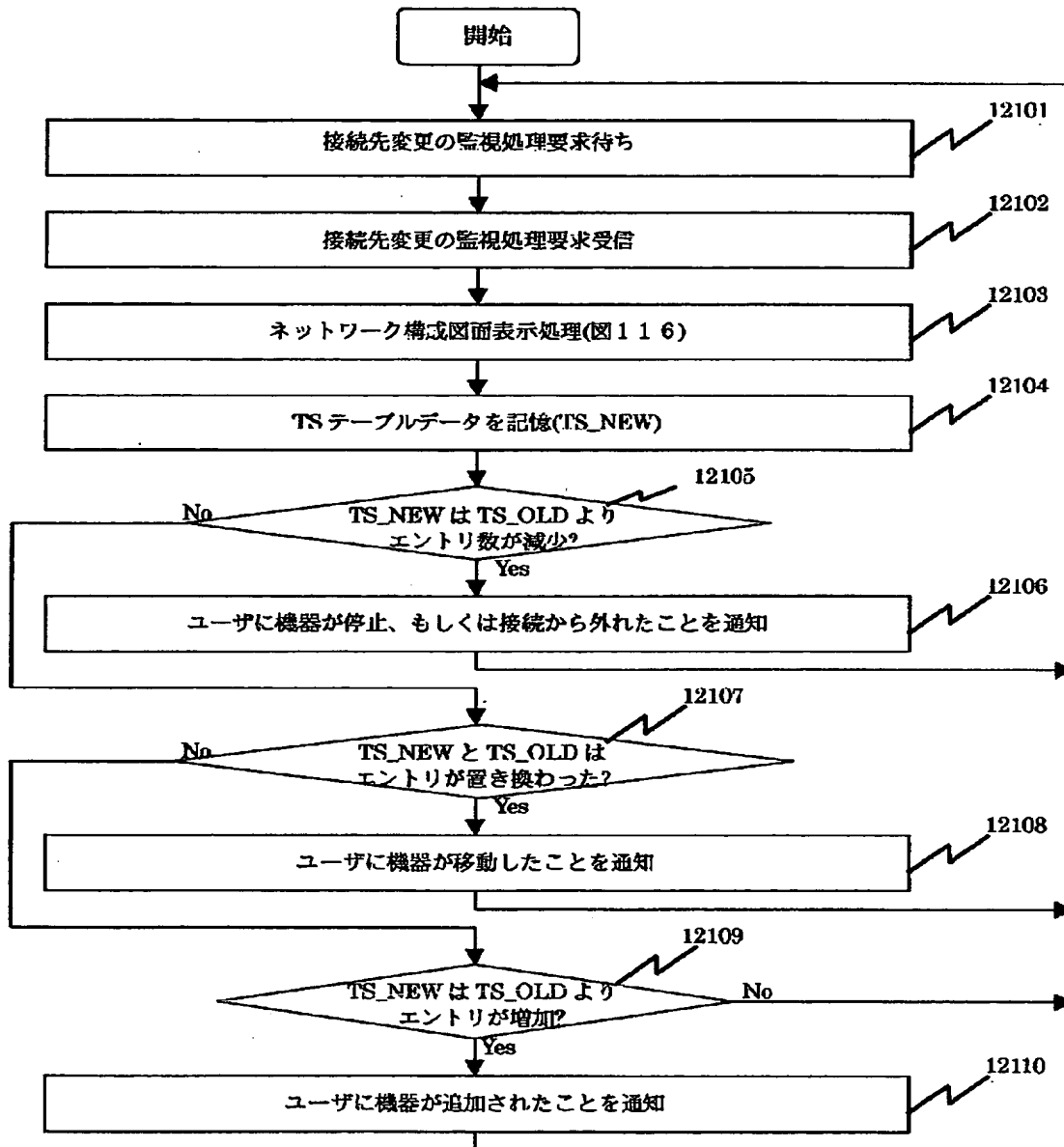
図面表示プログラムの動作フローチャート4
(Information 描画処理)



【図 1 2 1】

図 1 2 1

図面表示プログラムの動作フローチャート 5
(接続先変更の監視処理)



【書類名】 要約書

【要約】

【課題】 SNMPを実装しているインテリジェントなネットワーク機器が稼動しているネットワーク環境において、SNMP以外の特別なソフトウェアの実装を必要とせず、またSNMPの実装の仕方に依存せずに、ネットワークノード内部の物理的な機器構成を自動的に検出すること。

【解決手段】 SNMPマネージャを実装した管理者端末からネットワークノード内の各ネットワーク機器に対してICMPエコーリクエストを送信し、その応答によって稼動状態の機器を検出し、検出した各機器のSNMPエージェントに対し、当該機器内の管理情報ベースの格納情報の転送要求を送信し、返信された管理情報ベースの格納情報によってネットワークノード内に存在する機器の種別を検出する。ブリッジ機能を有する機器の管理情報ベースから各ポートに接続された機器の物理アドレスの集合を取得すると共に、ルーティング機能を有する機器の管理情報ベースから物理アドレスとIPアドレスの対応情報を取得し、その物理アドレスとIPアドレスの対応情報に基づき、ブリッジ機能を有する機器の各ポートの接続先の機器をIPレベルで認識する。

【選択図】 図 1 1

出 願 人 履 歴 情 報

識別番号

[000233055]

1. 変更年月日 1990年 8月 7日

[変更理由] 新規登録

住 所 神奈川県横浜市中区尾上町6丁目81番地

氏 名 日立ソフトウェアエンジニアリング株式会社